昭和58年10月3日第三種郵便物認可・1993年12月18日発行・毎月1回18日発行・通巻128号

MONTHLY The BASIC
[ザ・ベーシック]
ワンランク上をめざす人のパソコン情報誌
1994 no.128
1
January

ざべ

特集

サブノートの気になるあのこと



パソコン、もっと使いやすく――

BUFFALO



快適メモリ環境を実現する「MELWARE® Ver.5」

これからのWindows環境には、大容量メモリが必須。そのメモリ活用をより快適にするメモリ統合管理ソフト「MELWARE Ver.5」。快適Windows活用をトータルにサポートし、より使いやすいメモリ環境を構築します。メモリの性能はソフトで決まる。機能で、使いやすさで、定評のBUFFALO内部増設RAMボードをお選びください。

●Windows環境もDOS環境も、ひとつのドライバで統合管理。

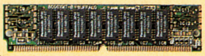
「MELWARE Ver.5」は、Windows3.1やMS-DOS環境で利用するさまざまなメモリ管理をトータルにサポート。ひとつのCONFIG. SYSで、Windows環境とDOS環境をリセットすることなく使い分けることが可能です。Windows環境への移行をスムーズにし、大容量メモリ活用時代に応えます。

●簡単設定で誰でもラクラクインストール。使いやすさが違う。

「MELWARE Ver.5」に誰にでも簡単にインストールできる、簡単設定プログラムを標準装備。CONFIG. SYSの変更などの面倒な作業を自動化。パソコンの使用環境(CPU、DOS、Windowsの有無)も自動で判別でき、目的や環境に合わせた各種設定がメニュー選択の簡単さでおこなえます。

NEC新型98MATE/98FELLOWに対応!

普及型 新98MATE/新98FELLOW専用
増設RAMサブボード
EFB NEW
合理的設計低コストタイプ



■PC-9821Bp、Bs、Be
PC-9801BA2、BS2、BX2用

4Mバイト(EFB-4M) ¥22,000
8Mバイト(EFB-8M) ¥43,000
16Mバイト(EFB-16M) ¥84,800
(MELWARE標準添付)

●パリティチェックが搭載されていない、PC-9821Bp、Bs、Be、PC-9801BA2、BS2、BX2専用とした低コストタイプ。

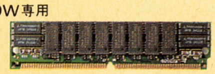
高機能型 新98MATE専用
内部増設RAMボード
EAC NEW



■PC-9821Ap2、As2専用
4Mバイト(EAC-4M)
…………¥30,000
(MELWARE標準添付)

●新増設RAMサブボード(EMF)で最大増設68Mまで(本体標準実装と合わせて73.6Mもしくは71.6Mまで)可能。
※Ap2/C9TにはNEC製ボード実装済。

新98MATE/新98FELLOW専用
増設RAMサブボード
EMF NEW
NEC製互換
新98MATE/新98FELLOW全機種対応



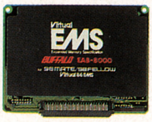
■PC-9821Bp、Bs、Be、Ap2※、As2※、Af※、Ts
PC-9801BA2、BS2、BX2用

4Mバイト(EMF-4M) ¥24,800 NEC製PC-9821A-B02L互換
8Mバイト(EMF-8M) ¥49,800 NEC製PC-9821A-B03L互換
16Mバイト(EMF-16M) ¥99,800 NEC製PC-9821A-B04L互換
(MELWARE標準添付) ※は専用内部増設RAMボードが必要です。

●64MB対応のMELWARE標準添付・メディアサイズをご指定ください。(3HD/5HD)

NECデスクトップ用内部増設RAMボード

■PC-9801BX、BA、PC-9821Ae、As、Ap専用
EAB



4Mバイト(EAB-4000T) ¥29,800
8Mバイト(EAB-8000T) ¥54,800
12Mバイト(EAB-12MT) ¥79,800
(MELWARE標準添付)
※XMCで追加増設14Mまで可能。

■PC-9801FA専用
EFA



4Mバイト(EFA-4000T) ¥29,800
8Mバイト(EFA-8000T) ¥54,800
(MELWARE標準添付)
※XMCで追加増設14Mまで可能。

■PC-9821Ce model S1/S2専用
ECE



4Mバイト(ECE-4000T) ¥29,800
(MELWARE標準添付)
※XMCで追加増設10Mまで可能。

■PC-9801FS、FX専用
EFS



4Mバイト(EFS-4000T) ¥29,800
(MELWARE標準添付)
※XMCで追加増設13Mまで可能。
(最終スロットにはXMB-1000T/1Mバイトを実装)

■PC-9801DA、RA21/51専用
EDA



4Mバイト(EDA-4000T) ¥29,800
8Mバイト(EDA-8000T) ¥54,800
(MELWARE標準添付)
※XMCで追加増設DAは12M、RA21/51は10Mまで可能。

★各パソコン用の内部増設RAMボードをラインアップ。

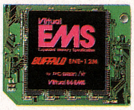
NEC対応内部増設RAMボードの追加増設に。
増設RAMサブボード
XMC



2Mバイト(XMC-2000T) ¥13,000
※NEC製PC-9801-61U/61F互換
※MELWAREは付属していません。

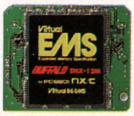
NEC98NOTE用内部増設RAMボード

■PC-9821Ne専用
ENE



4Mバイト(ENE-4000T) ¥26,000
8Mバイト(ENE-8000T) ¥48,000
12Mバイト(ENE-12MT) ¥69,800
(MELWARE標準添付)

■PC-9801NX/C専用
ENX



4Mバイト(ENX-4000T) ¥26,000
8Mバイト(ENX-8000T) ¥48,000
12Mバイト(ENX-12MT) ¥69,800
(MELWARE標準添付)

EPSONデスクトップ用内部増設RAMボード

■PC-486GR/GRスーパー/GR+/GF、386/S/G/V/GS/GE専用
ERB



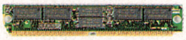
4Mバイト(ERB-4000S) ¥38,000
(MELWARE標準添付)
※XRB-4000S×2枚で追加増設12Mまで可能。

■PC-486SE/SR/P、386P専用
ERC



4Mバイト(ERC-4000S) ¥38,000
(MELWARE標準添付)
※XRB-4000S×2枚で追加増設12Mまで可能。

EPSON対応内部増設RAMボードの追加増設に
増設RAMモジュール
XRB



新486HX/HG対応
1Mバイト(XRB-1000S) ¥7,500 ※EPSON製/PCZRM2互換
2Mバイト(XRB-2000S) ¥13,000 ※EPSON製/PCZRM4互換
4Mバイト(XRB-4000S) ¥24,000 ※EPSON製/PCZRM5互換
(MELWARE標準添付)

※表示価格には消費税が含まれておりません。※Windowsは米国マイクロソフト社の商標です。※その他、掲載されている各製品名は一般に各社の商標または登録商標です。※デザイン、仕様等は改良のため予告なしに変更する場合があります。

株式会社 メルコ

●本社/〒460 名古屋市中区大須4-11-50 カミヤビル

▶お問合せはインフォメーションセンターまで
052-619-1827
受付/9:30～12:00・13:00～17:00〔月～金〕

資料請求No. 2

特　集

サブノートの気になるあのこと

最近，急に耳にするようになった，『サブノート』という新語には，さまざまな期待と不安が混在しているようにも見える．コンピュータが，より小さく，軽くなり，ますます身近になったように感じられる．今回は，このサブノートにスポットを当て，その可能性や求められる姿を探ってみることにした．また，実際に世間一般でサブノートと呼ばれるものを集め，その実力と使い勝手について検証を行ってみた．

MONTHLY The BASIC

新年特別インタビュー

キーマンに聞く－1994年の展望－

一般記事

TB TIME HOT NEWS

1

1994
January
no.128
CONTENTS

表紙デザイン：釣巻敏康
表紙イラスト：山崎のぶこ

連載

連載

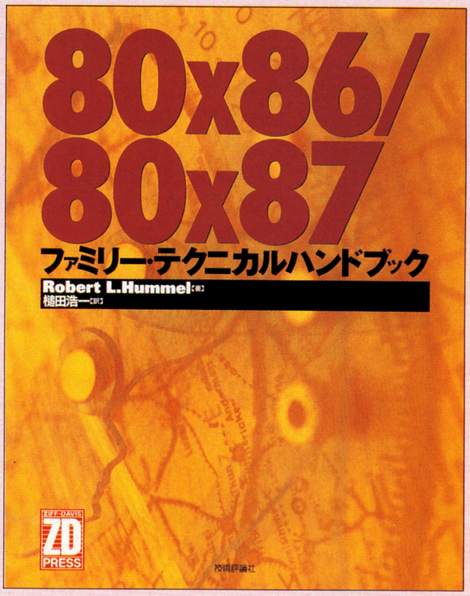
その他

パソコンライフを応援する技術評論社の書籍案内

80x86/80x87ファミリー・テクニカルハンドブック

Lobert L.Hummel 著，槌田浩一 訳
B5変形判・720ページ／定価5800円

パソコンに使用されるもっともポピュラーなCPUである，インテル製の80x86/80x87シリーズの完全なリファレンスブックです．初代8086をはじめ，最新の80486DXまでのプロセッサの仕組みと命令セットをくまなく掲載しているほか，インテルのマニュアルにも記載されていない未公開バグ情報や，プロセッサ間の相違についても解説しています．プログラマ必携の技術書です．



MS-C/C++
TURBO C++
BORLAND C++

実践入門C++

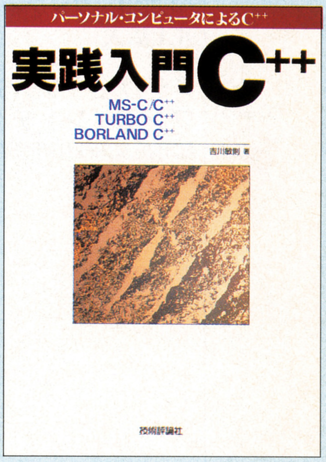
～パーソナル・コンピュータによるC++～

吉川敏則 著／A5判・320ページ／定価2400円

本書は中級ユーザーを対象としたC++の入門書です．C++ユーザーがつまずき易い点や，理解しにくい処理系上の機構，処理などについて，平易に解説しています．文法事項が手際良くまとめられていますので，理工系学生の教科書として，あるいはプログラマのサブマニュアルとして，重宝な１冊です．

【内容】

C++の基本入出力／クラスとは／クラスの継承／演算子のオーバロード他．



もよりの書店，大学生協，マイコンショップでお求め下さい．置いていない場合は，書名，冊数，定価，出版社名をお申しつけ下さい．

Cyrix
Advancing the Standards

かっとびサイリックス。



Upgrade Microprocessor Cx486

DR X2

2² SPEED!

■特長
- ●誰でも交換簡単386DXソケット互換、シングルチップ
- ●20／40MHz対応の独自倍クロック技術
- ●キャッシュ一貫性保証回路（米国特許出願中）
- ●クロック・デューティ補正回路

HERE

差し替えたその日から、あなたの386DX PCはかっとび486マシンに生まれ変わります。Windows環境が手に入り、これまでいらついていたグラフィックスがビックリするほど快適。サイリックスの業界初シングルチップ・アップグレードCPU：Cx486DRx2。超お手軽・超高速・超低価格のサイリックスかっとびテクノロジー、自信作です。



〈サイリックス販売特約店〉

サイリックスサポートセンター　TEL.03-3453-9168／関西電機　本社　TEL.082-877-5190／東京支社　TEL.03-5295-1161／札幌　TEL.011-261-0607／仙台　TEL.022-256-6431／名古屋　TEL.052-231-9952／大阪　TEL.06-266-1168／広島　TEL.082-264-3580／福岡　TEL.092-474-9571／アセットコア　TEL.03-3727-8401／アストロン　TEL.03-3255-3565／アークブレイン　TEL.03-3375-8968／メディアプリントサービス　TEL.03-3453-9021／ヒグチユニオン　TEL.03-3861-7388

〈サイリックス取扱店〉

亜土電子工業　T･ZONEパーツショップ　TEL.03-3257-2675／九十九電機　パソコン本店　TEL.03-3253-5599／パソコン本店II　TEL.03-3253-4199／DOS/Vパソコン館　TEL.03-3251-0099／ソフト8号店　TEL.03-3254-3999／名古屋2号店　TEL.052-251-3399／ツクモ札幌店　TEL.011-241-2299／DEPOツクモ2番街店　TEL.011-242-3199／通信販売部　TEL.03-3251-9911／上新電機　J&Pテクノランド　TEL.06-634-1211／J&Pメディアランド　TEL.06-634-1511／J&Pコスモランド　TEL.06-634-3111／J&Pビジネスランド　TEL.06-348-1881／J&P京都寺町店　TEL.075-341-4411／J&P大須店　TEL.052-262-1141／J&P渋谷店　TEL.03-3496-4141／J&P三宮1ばん館　TEL.078-231-2111／東映無線　テクノハウス店　TEL.03-3253-9896／ラジオセンター店　TEL.03-3253-0987／サンテク　東京営業所　TEL.03-3564-6561／岡山営業所　TEL.0862-41-1881／ダイワボウ情報システム　札幌支店　TEL.011-242-4231／新潟支店　TEL.025-287-5511／大阪支店　TEL.06-355-2454／新大阪支店　TEL.06-885-3200／高槻支店　TEL.0726-72-5200／神戸支店　TEL.078-392-2650／姫路支店　TEL.0792-88-6881／京都支店　TEL.075-823-2750／仙台支店　TEL.022-268-6131／千葉支店　TEL.043-225-7341／東京支店　TEL.03-3834-8391／新宿支店　TEL.03-5379-6531／横浜支店　TEL.045-471-0121／大宮支店　TEL.048-640-1985／名古屋支店　TEL.052-451-8251／岐阜支店　TEL.0582-65-5418／刈谷支店　TEL.0566-23-1131／浜松支店　TEL.053-456-0661／四日市支店　TEL.0593-52-0448／金沢支店　TEL.0762-63-2461／広島支店　TEL.082-238-3201／松江支店　TEL.0852-27-1621／出雲支店　TEL.0853-24-2668／岡山支店　TEL.0862-44-6408／高松支店　TEL.0878-65-0981／松山支店　TEL.0899-47-6379／福岡支店　TEL.092-262-7785／北九州支店　TEL.093-873-1650／佐賀支店　TEL.0952-22-2171／熊本支店　TEL.096-379-7131／長崎支店　TEL.0958-29-0266／東京ニーズ　本社　TEL.03-3832-3801／大阪営業所　TEL.06-391-7053／名古屋営業所　TEL.052-263-4811／仙台営業所　TEL.022-284-7681／テレビイ商会　TEL.03-3253-3861／フジオン　TEL.03-3255-7846／日本インコム・パソコン情報館　TEL.03-5951-2511／キョードー　TEL.03-3255-1751／ヘルスジャパン　TEL.03-3574-1151／東京トロン　TEL.03-5689-2888／システム・イン・山形　TEL.0236-42-1618／海外通商　TEL.052-263-1611／コンピューターPROショップ　TEL.0236-25-9801／コンピューターPROシステム店　TEL.0236-54-1531／ソービ　TEL.03-3780-3954

取扱店募集中！

資料請求No. 3

広告目次　ざべ　1 1994 No.128

広告掲載申込は㈱技術評論社メディアセンター03(3225)3481までお願いします

多数のMS-DOS用ソフトのキー入力を記録してリアルに再現します!

新発売!!



汎用リアル再現プログラム　*for PC-9801 series*

チェック&デモ

● 対応機種（本体メモリ640KB以上、ハイレゾモードを除く）
PC-9801シリーズ（PC-9801/E/F/M/U、LT、HAを除く）
PC-98XL/XL^2/RL/H98（XAを除く）
PC-9821シリーズ
PC-286・386・486シリーズ
MS-DOS Ver2.11以上（推奨Ver3.1以上）をご使用ください。

5"2HD/3.5"2HD　各¥24,000（税別）

●チェック&（アンド）デモの用途

日常業務

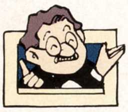


● ソフトの起動➡入力する➡別のソフトを起動・・・・といった定型キー操作を自動化できます。



多数のソフトで、汎用マクロとして活用OK!

教育・デモ



● 教育現場や仕事場で、目的ソフトのキー操作をあらかじめ記憶しておけば、いつでも再生OK。

● スライド等では表現しづらい連続画面や社内マニュアル等を、紙芝居感覚で自作できます。

ソフト開発



● リピート機能で、何度も同じ動作を繰返してデバッグの限界値テストが行なえます。

● チェック機能で、バグを発生させたキー操作手順を再現。記録レコード（最大60,000レコード）を参照して原因を究明できます。

●機能と特長

■汎用性
数多くのMS-DOS用ソフトに汎用マクロとして使えます。DOSコマンドや各種ソフトの起動操作OK! ソフト実行中に、いつでも記録や再生を開始できます。

■小サイズ
メモリ常駐プログラム。主メモリ使用量は約7KB（EMS時）、約40KB（非EMS時）。

■チェック機能
再生表示中の画面と保存画面のデータを比較照合。日付や時刻等、刻々変化する画面行をマスク（指定行を比較しないこと）できます。デバッグの省力化に!

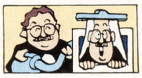
■リピート機能
同じ入力操作を何度も何度も繰返し再生。デバッグ時の反復動作チェックに!

■リアル再生
入力タイミングをリアルに再現。倍速、スロー等、再生スピードの調節OK!

■編集OK
記録データをテキスト変換。エディタで編集可。

■ポーズ（一時停止）機能
再生途中でポーズして、他のコマンド操作、画面取込み、差込み表示等がOK!

■画面取込み機能
ソフト実行中の画面をその場で取込みOK。
画面データは自動圧縮。デモや写真撮影に!



■日本語FEP対応
ATOK6、7、8／松茸3／VJE-β2、3／WXⅡ+に対応。

■純正マウス記録
純正マウスドライバ（MOUSE.SYS）のみ記録OK。

■短縮キー登録
日常作業を短縮キーに登録して、再生が簡単に行なえます。

※チェック&デモは、記録時と同じ環境で再生してください。動作環境の異なる場合、正常に動作しない場合があります。
カレントドライブやディレクトリを換えたりファイルの作成削除を行なった場合、記録時の状態に戻してからご使用ください。
※画面スクロール時等、キーを押しっ放しのままの状態で記録すると、キー操作が正確に再生できないことがあります。
※BIOSコールを使用しない等、独自のキー入力を行なうソフトでは使用できないことがあります。
※MS-WINDOWS、ジャストウィンドウ等、独自のキー入力を行なうプログラムには対応しておりません。



仮想編集型アセンブラ

DNA98

Disassembly Navigation Analyzer 98

● 対応機種（本体メモリ640KB以上、空きメモリ512KB必要、ハイレゾモードを除く）
PC-9801シリーズ（PC-9801/E/F/M/U、LT、HAを除く）
PC-98XL/XL^2/RL/H98（XAを除く）
PC-9821シリーズ
PC-286・386・486シリーズ
MS-DOS Ver2.11以上（推奨Ver3.1以上）をご使用ください。

5"2HD/3.5"2HD　各¥38,000（税別）

超高速、高精度、柔軟な逆アセンブル、仮想機能でリアルタイム編集を実現!

WINDOWS 3.0/3.1.2やOS/2のプログラムも逆アセンブルOK!!

DNA98は、従来の一括出力方式を一新した仮想編集機能を持つ超高速かつ高精度の逆アセンブラです。
DNA98は、編集、検索、表式等、ワープロにも匹敵する多彩な機能を備えています。
逆アセンブル専用エディタで自動生成した仮想ソースコードを、ラベル、コメントや命令セットの変更、コードデータ変換、検索、パッチ等、主メモリ上で自在かつ瞬時に編集できます。

DNA98の特長

■MS-WINDOWS3.0/3.1用プログラムOK
①WINDOWS3.0/3.1、OS/2（1.x/2.0）、MS-DOS用のASSEMBLER、FORTRAN、C等の実行ファイルの逆アセンブルOK。
②OBJファイル、メモリ内容、オーバーレイ、仮想/デバイスドライバもOK。

■80486命令対応
8088～486（16bit/32bitセグメント）、数値演算プロセッサの全命令セットに対応。

■超高速逆アセンブル
①50KBのファイルを1分弱、300KBのファイルでも6分前後で"超高速"自動逆アセンブル&自動編集。（当社測定値：80386、クロック20MHz、HDD、MS-DOSプログラム使用時）
②出力形式は、MASM4.0～6.0、TASM2.0等に準拠。

■高精度逆アセンブル
①最高19パスまでの"高精度かつ高速"逆アセンブル。
②ラベル・コメント自動生成。AT用/98用コメント選択や編集OK! ソースコードの日本語対応。

■大サイズプログラム対応
仮想逆アセンブル方式により、大サイズプログラムの逆アセンブルOK!
※部分逆アセンブル機能によりラベル制限を超える場合でも対応できます。なお、WINDOWS、OS/2のプログラムでは一部制限される場合があります。

■多彩な編集機能
DNA98の仮想機能により、瞬時に、
①自由自在なコード⇔データ変換、定数値の編集（定数⇔オフセット変換）、ラベル、コメントの編集や作成。
②MS-DOS用プログラムの仮想パッチ編集機能。
パッチの追加や削除、パッチ箇所の検索等がOK。

■強力な検索機能
多数の検索機能（テキスト、バイナリ、アドレス、コード・データ境界、ラベル、クロスリファレンス等）

■インタラクティブなユーザインターフェース
①プルダウンメニュー、ファンクションキーによるカーソル編集。
②マクロ一括編集、バッチ処理OK、マウス対応。（シリアルマウスを除く）

■アンパック・ユーティリティ添付
パックファイルやマルチセグメントCOMファイルを、EXE形式に自動変換。

＊MS-DOSシステムは内蔵していません。　＊広告記載の各製品は各開発メーカーの商標または製品名です。　＊製品のデザイン、仕様、価格等は予告なく変更することがあります。

マイクロデータ

〒169 東京都新宿区高田馬場1丁目17番8号
☎(03)3232-9801（代）

EPSON

先進の大容量フラッシュディスク
FLASH-PACKER



主な特長

- サンディスク社オリジナルの512バイトブロックでデータ消去できる専用フラッシュメモリチップを使用しており、ハードディスクと同様の管理を実現します。
- 高度な内部コントローラチップ採用により、ECC(エラー訂正)と書換え回数の長寿命化を実現します。
- 軽量、耐衝撃、耐振動に優れています。(耐衝撃性:1,000G動作時)
- 低電圧駆動設計による低消費電力化。

SDP5 SERIES

■JEIDA Ver4.1 (PCMCIA Rel. 2.0)準拠

機種名	容量	標準価格
SDP5-2.5	2.6MB	44,000円
SDP5-5	5.2MB	58,300円
SDP5-10	10.4MB	93,500円
SDP5-20	20.9MB	173,500円

■適応機種一覧表 (1993年11月4日現在)

メーカー名	機種名	備考
EPSON	PC-486 PORTABLE	—
HEWLETT PACKARD	HP95LX	*
〃	HP100LX	—
〃	OmniBook 300	—
日本IBM	ThinkPad 220	*
SOTEC	WiNBooK SX/33シリーズ	—
PROSIDE	HANDi-PC	—

*:該当機種のデバイスドライバーが別途必要です。
注)メーカー名、適応機種名は各社の商標または登録商標です。

国内代理店
エプソン販売株式会社 PD営業統括部

○お申し込みはエプソンOAサプライ㈱までお願い致します。TEL 03(3255)1528
○資料請求・技術的なお問い合わせについては
エプソン販売㈱メモリデバイス営業課までお願い致します。TEL 03(3377)7001

製造元
SUNDISK CORPORATION

---キリトリセン---

FAX注文書

- **お申し込み方法**…………下の注文書の太枠内に必要事項をもれなくご記入のうえ、FAXでご注文下さい。お電話でのご注文もお受け致します。
- **お届け方法**………………宅配便でお届け致します。
- **お支払い方法**……………お支払い方法は代金引換(商品代引)でお願い致しております。
- **商品の返品について**……商品の返品は到着後1週間以内にお願い致します。尚、開梱・開封後の商品の返品はお受けできない場合もございます。

あて先▶〒101 東京都千代田区外神田2丁目5番12号 タカラビル2F
エプソンOAサプライ株式会社 TEL.03(3255)1528

FAX.03(3258)7690

◉太枠内にご記入願います。

フリガナ お名前	TEL.	FAX.
ご住所 〒		

商品名	通販特価	数量	金額
SDP5-2.5	42,700円	個	円
SDP5-5	56,500円	個	円
SDP5-10	90,700円	個	円
SDP5-20	168,300円	個	円
		合計金額	円

※消費税は別途申し受けます。

Caravelle

DOS/V

Windowsをノートでも。DOS/Vノート、キャラベルから登場。

●高速 i486SX-25MHz ●軽量 1.8kg ●小型 B5+α サイズ



小型軽量の「B5NT」はフロッピードライブを外付けにすることにより、280×177×37mmのB5＋α サイズを実現しました。重量も1.8kgと軽量です。CPUにi486SX-25MHzを搭載し、640×480dotの16階調バックライト付き液晶ディスプレイで高速で鮮明な表示が得られます。また、ビルトイントラックボールを採用しましたので、Windowsの操作時もマウス接続の必要はありません。しかも 512KBのビデオRAMを持ち、外部にVGA対応ディスプレイを接続することにより、1,024×768(16色)の高精細表示が可能です。さらに、プリンタポート、シリアルポート、外部キーボード等の外部インターフェイスのほか、PCMCIA Type2のカードスロットと、120pinのISAバスコネクタを装備し、拡張性も十分。FAXモデムカードなどの実装も可能です。メモリーを標準で4MB(最大20MBまで増設可能)搭載しましたので、メモリーの増設なしでWindowsが動作します。ハードディスクは80MB・120MBの脱着が可能なパック式を採用しましたので、アプリケーションごとにハードディスクを使い分けるなど、利用方法はアイデア次第で無限大。バッテリーはNi-Cdタイプを使用し、バッテリーの残量が少なくなると、自動的に省電力モード(通常の1/8の消費電力)に切り替わります(設定による)ので、安心して作業をすることができます。レジューム機能を搭載していますので、作業の中断／再開が容易です。

AV-B5NT Series

●B5 NT Series　(外付けFDDユニット・ACアダプタ・キャリーバッグ・バッテリー付)
- ·AV-B5 NT- 80(メモリー4MB· 80MBHDD内蔵)······¥220,000
- ·AV-B5 NT-120(メモリー4MB·120MBHDD内蔵)······¥230,000
- ·AV-B5 NT-200(メモリー4MB·200MBHDD内蔵)······近日発売

●B5 NT Series (外付けFDDユニット・ACアダプタ・キャリーバッグ・バッテリー付・MS-DOS5.0J、日本語MS-Windowsインストール済み)
- ·AV-B5 NT- 80W(メモリー4MB· 80MBHDD内蔵)·····¥238,000
- ·AV-B5 NT-120W(メモリー4MB·120MBHDD内蔵)·····¥248,000
- ·AV-B5 NT-200W(メモリー4MB·200MBHDD内蔵)·····近日発売

AV-B5NT Series用オプション

製品名	価格	製品名	価格
RAMカード (4MB)	¥ 43,000	外付け3.5"FDD	¥ 22,000
RAMカード (16MB)	¥210,000	外付け3.5"FDD用ケーブル	¥ 2,500
バッテリーパック	¥ 7,800	ドッキングステーション	¥ 45,000
専用内蔵 80MBハードディスク	¥ 49,000	AC/DCアダプター	¥ 14,000
専用内蔵120MBハードディスク	¥ 59,800		

		B5NT
CPU		i486SX-25MHz
メモリー	標準	4MB
	最大	20MB
表示装置		CFLバックライト付きFSTN LCD (640×480) 16階調
外部表示装置		VGAコンパチブル
		CRT:1024×768(16色) 800×600(256色)
記憶装置		ハードディスク(IDE)パック内蔵
		外部FDD3.5インチ 2モード(720KB／1.44MB)
キーボード		日本語 86 Keyboard
入出力ポート	キーボード	IBM仕様インターフェイス
	マウス	PSマウス互換
	パラレル	セントロニクス準拠×1
	シリアル	RS232-C準拠×2
	CRT	VGAタイプ
	外部FDD	3.5インチ 2モード(720KB／1.44MB)
	カードスロット	PCMCI Type2
	拡張バス	120pin ISAバス
サイズ		280(W)×37(H)×177(D) mm
消費電力(最大)		20W (ACアダプタ使用時)
動作温度・湿度		0～45°C 8～80%RH
重量		1.8kg

※商品の価格には消費税は含まれておりません。※記載の商品名は各社の商標または登録商標です。　※製品改良の為、仕様·意匠·外観等は予告なく変更することがあります。

Caravelle Data System

製造元　株式会社キャラベルデータシステム
本社　〒107東京都港区赤坂2-18-3 (三葉ビル)
TEL(03)5561-9361(代) FAX(03)5561-9360
仙台営業所　〒980宮城県仙台市青葉区二日町6-13(仙台グリーンコーポ二日町ビル)
TEL(022)267-5524 FAX(022)267-5526
福岡営業所　〒812福岡県福岡市博多区比恵町2-7博多東エースビル6F
TEL(092)482-2581 FAX(092)482-2571

販売元　シーディーエス販売株式会社
本社　〒107東京都港区赤坂2-18-3 (三葉ビル)
TEL(03)5561-0171(代) FAX(03)5561-0170
大阪営業所　〒540大阪府大阪市中央区内本町1-2-13(谷四ばんらいビル)
TEL(06)943-5543(代) FAX(06)943-6810

資料請求No. 7

TB TIME HOT NEWS Part 1

NEC新製品発表

98新ラインアップ,普及型98MATE,高機能型98MATEと新98FELLOW

11月10日，NEC(03-3452-8000)は98FELLOW/MATEシリーズを一新し，機能向上を図りながら低価格化を実現した「98FELLOW」3機種，「98MATE」5機種などPC-9800シリーズ計10機種を発売すると発表した．



■普及型98MATE　PC-9821Bs

PC-9821Bp/Bs/Be

98MATEシリーズには，現行モデルからPCM音源や拡張ローカルバススロットなどを省いた「普及型98MATE」，PC-9821Be/Bs/Bpの3機種を発売．PC-9821Ae/As/Apに比べ，約40～45%の低価格化を実現した．

同シリーズは，HDDモデルのみの構成で，全モデルにWindows3.1がプリインストールされる．また，マザーボードにCIRRUS LOGIC社のグラフィックアクセラレータチップCL-GD5428をローカルバス接続し，Windows上で640×480ドット/1,677万色，1,024×768ドット/256色表示を実現している．ただし，従来機種ではサポートされていたMS-DOS上の640×480ドット/256色表示はサポートされていない．

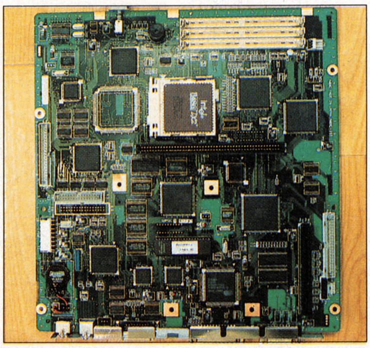
メモリは標準で5.6MB搭載し，Windowsの利用に備えている．現行の98MATEではメモリは最大14.6MBまでしか増設できなかったのに対し，本体内部の72ピンSIMMソケットに最大35.6MBまで増設可能．このほか，Pentiumオーバードライブプロセッサ(P24T)に対応．PC-9821Bpには128KBのセカンドキャッシュメモリがオプションで用意される．

FDDは3.5インチ3モードドライブを1基のみ搭載．外付け用1MB FDD I/Fは持っていない．このため，FDD2台を必要とするアプリケーションソフトにも対応できるように「仮想FDドライブ機能」を装備している．また，コピープロテクトなどのため，仮想FDドライブ機能を使用できない場合に備え，増設フロッピィドライブベイを用意．2台目のHDDやCD-ROM用にはファイルベイ(従来のファイルスロットとは別もの．I/FはNEC公表では専用I/Fだが，実体はIDE)が用意されている．

なお，マウスのコネクタ形状はD-SUB 9ピンからミニDIN9ピン(H98と同じ)に変更されるとともに位置も本体背面に変更された．パラレルI/Fも双方向のフルセントロニクスになり，形状が変わった(変換アダプタ付き)．

■PC-9821Bp/Bs/Beの基本仕様

	PC-9821Bp	PC-9821Bs	PC-9821Be
CPU/クロック	i486DX2/66MHz	i486SX/33MHz	i486SX/25MHz
セカンドキャッシュ	128KB(別売)	−	−
メインメモリ(最大)	5.6MB(35.6MB)		
VRAM	テキスト12KB/グラフィック1MB		
FDD	3.5インチ(1.44MB/1.2MB/720KB)×1		
HDD	170MB/340MB	170MB	120MB
アクセラレータチップ	CIRRUS LOGIC CL-GD5428(ローカルバス接続)		
インターフェース	RS-232C(最大19,200bps)，ディスプレイ，キーボード，パラレルI/F(プリンタI/F変換アダプタ付き)，マウス		
汎用拡張スロット	×3		
ファイルベイ	×1		
電源	100V(50/60Hz)		
最大消費電力	120W		
外形寸法(mm)	380(W)×335(D)×150(H)		
重量(kg)	7.7		
価格	39万8千円/43万8千円	39万8千円	26万8千円



■PC-9821Bpのマザーボード．CIRRUS LOGIC社のCL-GD5428が搭載されたほか，メモリ増設用に72ピンのSIMMソケットが追加された．



■高機能型98MATE　PC-9821As2

PC-9821Ap2/As2

98MATEの機能強化モデルとしては，HDD（Windows3.1プリインストール）モデルのマザーボードにS3社のグラフィックアクセラレータチップ86C928をローカルバス接続（VRAMは1MBで増設はできない）．最上位機種のPC-9821Ap2/C9Tには，現在業界最高速のグラフィックアクセラレータチップであるMatorox社のMGA2を搭載した32ビットローカルバス専用「ウィンドウアクセラレータボードA2」（VRAM 2MB，11万円）を拡張スロット#4に標準装備する．また，セカンドキャッシュメモリを最大256KBまで搭載可能とし，Ap2には標準で128KBのセカンドキャッシュメモリが搭載されているなど，グラフィック機能の強化と高速処理を実現している．

メモリは，FDDモデルが3.6MB，HDDモデルが5.6MB，最上位機種のAp2/C9Tは7.6MBを標準装備．FDDモデルとAp2/C9Tは最大71.6MB，HDDモデルは73.6MBまで増設可能となり，WindowsNTなどの次世代OSへの対応も考慮されている．また，ファイルスロットのほか，増設フロッピィディスクドライブベイを装備．Pentiumオーバードライブプロセッサ（P24T）にも対応している．

なお，PC-9821Ap2/C9W，Ap2/C9Tには，PhotoCD（マルチセッション）対応の倍速CD-ROMドライブがファイルスロットに標準搭載され，本体SCSIスロットにはPC-9821A-E10が標準装備される．なお，最上位モデルAp2/C9Tのみ12月27日出荷予定．

また，同モデルにはN5200シリーズに採用されているOS「PTOS」をプリインストールしたPC-9821As2/U7P，Ap2/U8Pが用意され，企業内情報システム端末にも対応する．

98FELLOW

98FELLOWもラインアップを一新．CPUにi486SX/25MHzを搭載したローエンドモデルPC-9801BX2/U2が17万8千円と従来機種よりも約18%の低価格化を実現している．メモリは最大19.6MBまで増設可能（PC-9801BX2/U2,BX2/M2では17.6MB）．BA2にはオプションで128KBのセカンドキャッシュが用意される．また，普及型98MATE同様，FDD2台を必要とするアプリケーションソフトにも対応できるように「仮想FDドライブ機能」を装備．また，コピープロテクトなどのため，仮想FDドライブ機能を使用できない場合に備え，増設フロッピィドライブベイが用意され，2台目のHDDやCD-ROM用にはファイルベイが用意されている．



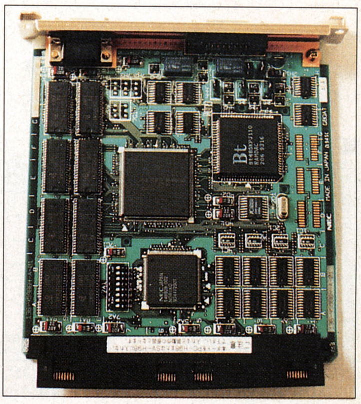
■PC-9801BX2

PC-9821Ts

PC-9821Tsは，PC-9801Tの後継機種にあたり，4,096色中256色表示可能な640×400/480ドットの9.5インチTFTカ

■PC-9821Ap2の基本仕様

	PC-9821Ap2		
	Ap/U8W	Ap/C9W	Ap/C9T
CPU/クロック	i486DX2/66MHz		
セカンドキャッシュ	128KB（最大256KB）		
メインメモリ（最大）	5.6MB（73.6MB）		7.6MB（71.6MB）
VRAM	テキスト12KB/グラフィック512KB		
（アクセラレータ専用）	1MB		2MB
アクセラレータチップ	S3 86C928（ローカルバス接続）		MATROX MGA2
ウィンドウアクセラレータ	本体に内蔵		ウィンドウアクセラレータボードA2
サウンド機能	PCM音源2チャンネル,FM音源6和音,リズム音源6和音,SSG音源6和音		
FDD	3.5インチ（1.44MB/1.2MB/720KB）×1		
HDD	340MB	510MB	
CD-ROM	–	内蔵：倍速,PhotoCD（マルチセッション）対応	
インターフェース	RS-232C（最大19,200bps），ディスプレイ，キーボード，マウス，パラレルI/F（プリンタI/F変換アダプタ付き），オーディオ入出力		
SCSI I/F	–	PC-9821A-E01相当品内蔵	
汎用拡張スロット	×4（内2スロットは32ビットローカルバスコネクタを装備，U8Wは拡張スロット#4にウィンドウアクセラレータボードA2を実装済み）		
ファイルスロット	×1		
電源	100V（50/60Hz）		
最大消費電力	220W		
外形寸法（mm）	380（W）×335（D）×150（H）		
重量（kg）	10.2	11.4	11.7
価格	58万円	70万円	80万円



■Matrox社の最新型アクセラレータチップMGA2を搭載した「ウィンドウアクセラレータボードA2」．MGA2は，世界最速のアクセラレータチップとして評判のMGAから3Dグラフィックエンジンなどを省いた廉価版製品．Matrox社では，日本国内においては同チップをNECに独占供給すると言われる．

惜しむらくは，今回も800×600ドットおよび6万色ドライバがサポートされていないこと．

ラー液晶を採用．Windows環境に応える省スペース・省エネルギーパソコンとして位置付けている．

同モデルは，CPUにi486SX/33MHzを搭載，メモリはHDDモデルで5.6MB（最大39.6MB），FDDモデルは3.6MB（最大37.6MB）を標準装備．マザーボードには，CIRRUS LOGIC社のCL-GD5428をローカルバス接続し，CRTを用いることにより640×480ドット/1,677万色，1,024×768ドット/256色表示をサポート．拡張スロットも2スロット装備している．

価格は，120MB HDDを搭載し，Windows3.1をプリインストールしたPC-9821Ts120/Wが77万円．FDDモデルPC-9821Tsが67万円．



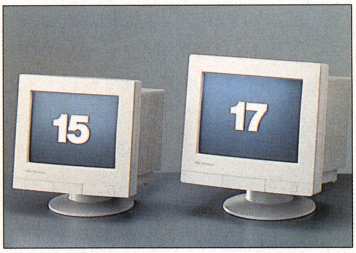
■PC-9821Ts

なお，今回発表された98MATE/98FELLOW/PC-9821Tsは，PC-9821Af同様，RS-232C最大19,200bpsに対応．さらに，全機種パワーマネージメント機能をサポート．一定時間入力がない場合，本体内蔵のFDDやHDDのモータを自動的に停止するほか，今回同時に発表されたVESA規格のパワーマネージメント機能搭載のマルチシンクディスプレイ「PC-KM152/172/211」（13万8千円/19万8千円/58万円）と組み合わせることにより，Windows 3.1使用時に，一定時間入力がない場合は，約47%の低電力化を実現する．

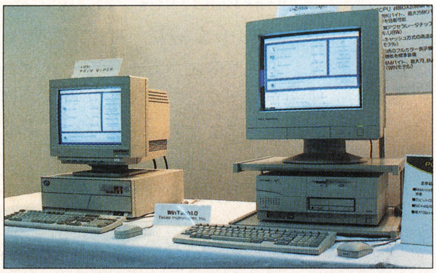
また，PhotoCDマルチセッション対応の本体内蔵型倍速CD-ROMドライブ（愛称マルチスピン）を発表．98FELLOW，普及型98MATEのファイルベイ内蔵用のPC-CD60D，ファイルスロット内蔵用PC-CD60Fとも5万8千円．

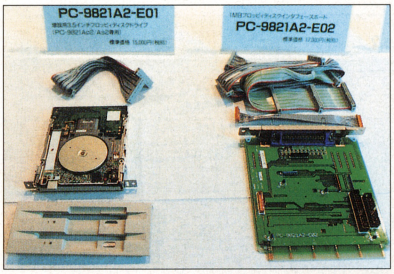
■PC-9801BA/BS/BXの基本仕様

	PC-9801BA2/U7	PC-9801BS2/U7	PC-9801BX2/U7
CPU/クロック	i486DX2/66MHz	i486SX/33MHz	i486SX/25MHz
セカンドキャッシュ	128KB（別売）	－	－
メインメモリ（最大）	3.6MB（35.6MB）		
VRAM	テキスト12KB/グラフィック256KB		
FDD	3.5インチ（1.44MB/1.2MB/720KB）×1		
HDD	120MB		
アクセラレータチップ	－		
インターフェース	RS-232C（最大19,200bps），ディスプレイ，キーボード，パラレルI/F（プリンタI/F変換アダプタ付き），マウス		
汎用拡張スロット	×3		
ファイルベイ	×1		
電源	100V（50/60Hz）		
最大消費電力	120W		
外形寸法（mm）	380（W）×335（D）×150（H）		
重量（kg）	7.5		
価格	33万8千円	27万8千円	23万8千円

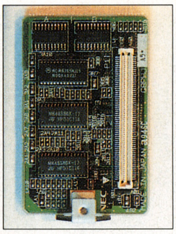


■PC-KM152とPC-KM172


■会場では，IBM PS/V搭載のVIPER VLB(Weitek POWER9000)とMGA2チップを搭載したAp2のベンチマーク比較が行われた．

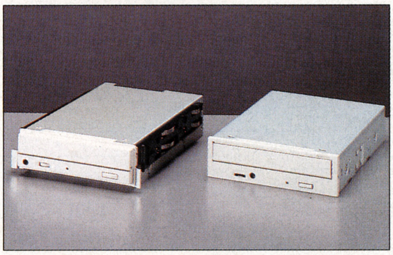


■増設用フロッピィディスクドライブベイに増設可能な3.5インチFDD(左)と1MBフロッピィディスクインターフェースボードPC-9821A2-E02．なお，増設用FDDも98MATEと普及型MATE/FELLOWで異なる．


■セカンドキャッシュ



■256色表示と1,677万色表示との比較も行われた．


■PhotoCD(マルチセッション)対応の倍速CD-ROMドライブ．左がファイルスロット用(PC-CD60F)，右がファイルベイ用(PC-CD60D)．

TB TIME HOT NEWS Part 2

IBM新製品発表

CD-ROM内蔵マルチメディア対応 PS/V Vision,BlueLightning搭載機など

日本アイ・ビー・エム㈱(0120-04-1992)は，11月4日，倍速CD-ROMや1,677万色表示など，マルチメディアに最適なディスプレイ一体型パソコン「PS/V Visoin」とDSTN方式のカラー液晶ディスプレイを採用した「ThinkPad」シリーズ2機種を発売すると発表した．

PS/V Vision

「PS/V Vison」は，CPUにIBM 486SLC2/50MHzを搭載，メモリ8MB，HDD 170MB(もしくは340MB)，倍速CD-ROMドライブ，SoundBlaster16を標準装備．IBM DOS J5.0/VとWindows 3.1がプリインストールされる．

14インチのCRTディスプレイ一体型の比較的コンパクトな本体には，ATバススロットが3基用意され，内1スロットをVL-Busスロットとして使用することができる(ATバススロット1基はSoundBlaster16を装着済み)．また，同モデルはマルチメディアパソコンの規格であるMPC Level2*に準拠．さらにCIRRUS LOGIC社のグラフィックアクセラレータチップCL-GD5428をローカルバス接続することにより，640×480ドット/1,677万色，1,024×768ドット/256色の表示を可能としているほか，松下電器との共同開発によるアンプ内蔵のステレオスピーカを標準装備する．価格は，HDD 170MBモデルが36万8千円．340MBモデルが43万8千円．

なお，12月末日までのキャンペーン期間中は，米Lucas Arts社のSTAR WARSを題材とした最新3Dシューティングゲーム「REBEL ASSAULT」の体験版ほか5枚のCD-ROMソフトが付属する．



■PS/V Vision

ThinkPad750Cs/330Cs

また，「ThinkPad750C」のディスプレイを9.5インチデュアルスキャンSTNカラー液晶に換え，低価格化を図った「ThinkPad 750Cs」(60万8千円〜)と，IBM 486SLC2/50MHzを搭載し，「ThinkPad750Cs」と同様，9.5インチDSTN液晶を採用したエントリーモデル「ThinkPad330Cs」(34万8千円〜)の2機種の発売も同時に発表された．

「ThinkPad750Cs」は，CPUにi486SL/33MHzを搭載．本体にPCM音源チップを標準装備し，マルチメディア環境で必要なデジタル．サウンド機能として高品質サウンドの録音・再生が可能．また，海外でも保証が受けられる「海外旅行者向け用ハードウェア

■PS/V Visionの基本仕様

	PS/V Vision	
	2408-WMB	2408-WMC
CPU/クロック	IBM 486SLC2/50MHz	
メインメモリ(最大)	8MB(16MB)	
VRAM	1MB	
FDD	3.5インチ3モード(1.44MB/1.2MB/720KB)×1	
HDD	170MB	340MB
CD-ROM(トレイ式)	倍速PhotoCDマルチセッション対応	
オーディオ機能	16ビット44.1kHz PCM音源(SoundBlaster16)	
拡張バス	ATバス×3(うち1スロットはSoundBlaster実装)	
インターフェース(ポート)	RS-232C(D-Sub9ピン)×2，プリンタ，キーボード，マウス，ディスプレイ	
ビデオサブシステム	CIRRUS LOGIC CL-GD5428	
ディスプレイ	ドットピッチ0.28mm 14インチフラット管	
キーボード	JIS配列106キー日本語キーボード	
マウス	○	
DOS 5.0/V	○	
Windows 3.1	○	
OS/2 J2.1	−	
電源(最大消費電力)	AC-100-120V(230W)	
外形寸法(mm)	319(W)×448(D)×432(H)	
重量(kg)	18	
価格	36万8千円	43万8千円



■発表会場では，98MULTiやMacintoshとの比較デモが行われた．

*MPC Level2 ：Multimedia PC Marketing Councilが定めるマルチメディアパソコンの規格で，以下の条件を満たすもの．CPU:486SX/25MHz，メモリ:4MB，HDD:160MB，

保証(3年間)」の対象になる.

「ThinkPad330Cs」は，CPUにIBM 486SLC2/50MHzを搭載．PCMCIA2.1/JEIDA Ver.4.1準拠のPCカードTypeⅡスロットを1基装備．従来別ユニットだったACアダプタを本体に内蔵することにより，本体に直接電源ケーブルを接続することを可能にしている．このACアダプタは着脱可能．バッテリパックは別売．

なお，両モデルにはPCカードのセットアップ作業を大幅に軽減し，PCカードの情報表示やカード挿入時にアプリケーションを自動的に起動させることができる「Easy Playing」機能が標準搭載される．

また，同時にPS/VのOS/2 J2.1プリインストールモデルなど2機種を追加と一部機種の値下げも発表された．

BlueLightning搭載PS/55

同社は，11月24日，IBM BlueLightning 486BL2/66MHzを国内モデルとしてははじめて採用したPS/55シリーズの新製品「PS/55 モデル5550-L」を発売すると発表した．

IBM 486BL2は，i486DX2と互換性を持ち，32ビットアドレスと16KBのキャッシュメモリを搭載，i486DX2/66MHzとほぼ同等のパフォーマンスを発揮する．ハードディスク用インターフェースにはIDEを採用．MicroChannelスロット×4，ストレージベイ×1を装備する．

価格は，メモリ4MB，HDD 250MB，XGA-2搭載のIBM DOS J5.0/Vプリインストールモデルの「5551-LXC」が44万5千円．この他に，倍速CD-ROM標準搭載モデル(5551-LEC)とIBM DOS J5.0プリインストールモデル(5551-LOC)が用意される．

■ThinkPad330Csの基本仕様

	ThinkPad 330Cs			
	J11	JB8	JBB	JBW
CPU/クロック	IBM 486SLC2/50MHz			
メインメモリ(最大)	8MB(12MB)			8MB(8MB)
VRAM	512KB			
FDD	3.5インチ(2.88MB/1.44MB/1.2MB/720KB)×1			
HDD(着脱式)	–	80MB	170MB	
PCカードスロット	PCMCIA TypeⅡ/Ⅰ×1			
DRAMカードスロット	1			–
インターフェース(ポート)	RS-232C(9ピン)，プリンタ，ディスプレイ キーボード/マウス，拡張バスコネクタ(AT)			
ビデオサブシステム	VGA			
ディスプレイ	9.5インチDSTNカラー液晶 256色			
DOSJ5.0/V	–	○	○	○
Windows J3.1	–	–	–	○
電源	ACアダプタ 100～125V(50/60Hz) バッテリパック(オプション)充電時間 約2時間 連続動作時間 約2.0～6.0時間			
最大消費電力	21W			
外形寸法(mm)	297(W)×210(D)×51.5(H)			
重量(kg)	2.4	2.6		
価格	34万8千円	41万8千円	44万8千円	49万8千円



■ThinkPad330Cs

■PS/55モデル5550-Lの基本仕様

	PS/55 モデル5550-L		
	LOC	LXC	LEC
CPU/クロック	IBM 486BL2/66MHz		
キャッシュメモリ	16KB(CPU内蔵)		
メインメモリ(最大)	4MB(28MB)	8MB(32MB)	
VRAM	1MB		
FDD	3.5インチ(2.88MB/1.44MB/1.2MB/720KB)×1		
HDD(IDE)	250MB(最大590MB)		
インターフェース(ポート)	RS-232C(9ピン)×1/(25ピン)×1，プリンタ，ディスプレイ，キーボード，マウス		
バスアーキテクチャ	Micro Channel		
拡張スロット	4		3
ビデオサブシステム	PS/55日本語表示	XGA2	
IBM DOSJ5.0/V	–	○	○
IBM DOSJ5.0	○	–	–
CD-ROM	–	–	○
電源	ACアダプタ 100～125V(50/60Hz)		
最大消費電力	110W		
外形寸法(mm)	360(W)×415(D)×103(H)		
重量(kg)	9.0		
価格	45万5千円	44万5千円	54万5千円



■PS/55 モデル5550-L

CD-ROM:倍速/PhotoCD対応，AUDIO:16ビットステレオ，グラフィック：640×480ドット/6万色，OS:Windows3.1

TB TIME HOT NEWS Part 3

EPSON新製品発表

98フルスペックをA5サイズに凝縮 EPSON PC-486PORTABLE

セイコーエプソン㈱(03-3377-3531)は，11月4日，98アーキテクチャをA5ファイルサイズに凝縮し，携帯性を追求したアップグレードPC「PC-486 PORTABLE」をエプソン販売㈱を通じて12月中旬より発売すると発表した．

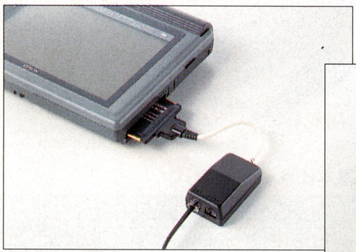
同製品は，CPUにSLエンハンスi486SX/25MHzを採用，SLエンハンスi486DX2/40MHzへのアップグレードも可能．メモリは標準で3.6MB(最大11.6MB)，PCMCIA2.01/JEIDA Ver.4.1準拠のTypeⅡスロットおよびTypeⅢスロットを各1基搭載．また，TypeⅢスロットには，オプションの1.8インチハードディスクカード(40MB:5万8千円/80MB:7万8千円)が装着可能．ディスプレイは，ドットピッチ0.23mmの9.6インチFTN液晶を採用．ROMにPen DOS2.10およびMS-DOS Ver.5.0の基本機能，ビジネスマネージャ(BM)がインストールされ，ペン入力パソコンとしても使用できる．



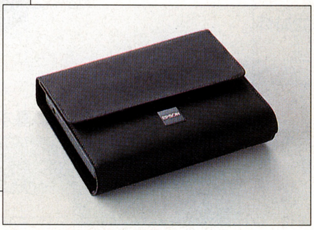
■PC-486PORTABLE

ビジネスマネージャは，プランリスト，アクションリスト，スケジューラ，住所録，電話帳，ノート(図形・文字)，アクセサリ(メモ・電卓・世界時計)などの機能をペン入力で使用できるもの．この機能は，98アプリケーションを実行中でもワンタッチで呼び出すことができる．

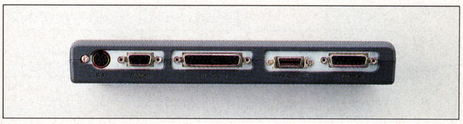
本体重量は900g，これにキーボード(システム手帳タイプのケースつき)を装着した状態では1.6kgとなる．このほかに，3モード対応の3.5インチ外付けFDDとRS-232C，プリンタ，マウス，キーボード，CRTなどの接続を可能にする拡張コネクタBOXが標準で添付される．価格は28万8千円．



■オプションのFAX モデムカード



■本体はシステム手帳タイプのケースに収まる．



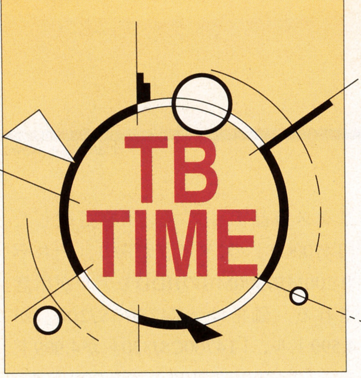
■拡張コネクタBOXには，左からキーボード，マウス，RS-232C，プリンタ，アナログRGBコネクタが用意され，柔軟な拡張性を持つ．

■PC-486PORTABLEの基本仕様

	PC-486 PORTABLE
CPU/クロック	i486SX/25MHz
メインメモリ(最大)	3.6MB(11.6MB)
ディスプレイ	モノクロFTN液晶 640×400ドット/8階調
FDD(外付け)	3.5インチ(1.44MB/1.2MB/720KB)×1
インターフェース	RS-232C，ディスプレイ，キーボード，マウス，プリンタ，FDD
ICカードスロット	PCMCIA 2.0/JEIDA Ver.4.1準拠 TypeⅡ×1，TypeⅢ×1
電源	NiCdバッテリまたはACアダプタ
バッテリ駆動時間	標準バッテリ：3時間/大容量バッテリ：5時間
最大消費電力	19W
外形寸法(mm)	234(W)×175(D)×48(H)
重量(kg)	1.6
価格	28万8千円



■CPUやPCMCIA 2.0 / JEIDA Ver.4.1準拠のICカードスロットなどは，すべてこの液晶の下に収められている．



Micro Device

MicroDevice

AMD，ワンチップのイーサネットコントローラ

日本AMD㈱(03-3346-7560)は，同社のマザーボード用ワンチップイーサネットコントローラ製品「PCnet」ファミリの新製品3機種を発売した．

Am79C970(PCnet-PCI)とAm79C965(PCnet-32)は，PCIとVL-Busをサポート．Am79C961(PCnet-ISA+)はMicrosoft社が提唱するISA Plug and Playに準拠したはじめてのイーサネットコントローラ．また，同社はIC以外にもドライバソフトウェア一式も準備している．サンプル価格はAm79C970 (PCnet-PCI:132ピン PQFP)とAm79C965 (PCnet-32:160ピン PQFP)が3,750円，Am79C961(PCnet-ISA+:132ピン PQFP)が3,250円．

MicroDevice

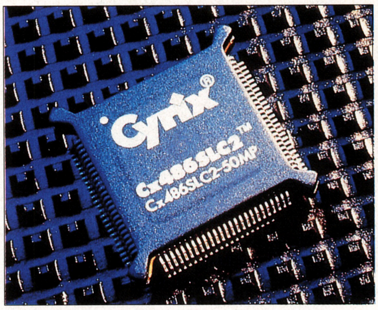
Cyrix，倍クロック機能搭載Cx486SLC2

■Cx486SLC2

Cyrix Corporation(日本法人：サイリックス㈱045-471-1661)は，11月15日に，COMDEX/Fall '93にて「Cx486SLC2-50」を発表した．

Cx486SLC2は，Cx486SLC/eの倍クロック機能拡張版．小型の100ピンMQFPパッケージに封止されたCx486SLC2は，従来のCx486SLC/e同様，電源管理機能，スタティック動作，システム管理機能，動的クロック停止機能などをサポートしている．内蔵キャッシュの容量は，Cx486SLC/eと同様に1Kバイトとなっている．Cx486SLC2-50は，外部バス25MHz動作の高速50MHz CPUコアの採用で，Cx486SLC-33と比較しても最大50%高速となっている．





■Cx486SLC2

また，COMDEX/Fall '93において，EPSONをはじめとしたいくつかのメーカーから，Cx486SLC2-50を採用したノートブックPCが発表されている．

Cx486SLC2-50 5V版は，現在量産中で，OEM価格は千個購入時で9,800円．

ヒグチユニオン㈱(03-3861-7388)では，Cx486SLC2の換装代行業務を行う．換装代行料金は5万円より．保証は改造後1年間は無保証となっている．

■Cyrix Instead

Cyrix Corporationは，米国内でIntel製品からの買い換え需要の誘発を狙った"Cyrix Instead"(サイリックスを代わりに)というキャンペーンに乗り出した．日本国内でのキャンペーン実施については，現在検討中とのこと．

MicroDevice

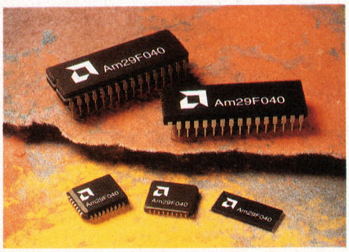
ソニー，JEIDA Ver.4.1準拠のフラッシュメモリカード

ソニー㈱(03-5448-2917)は，JEIDA Ver.4.2/PCMCIA 2.1に準拠したICメモリカード「IMS01MA」,「IMF02MA1」を発売する．「IMS01MA」は高速な書き込みが可能なSRAMチップを内蔵し，1MBのデータを保存することができ，頻繁に書換が必要なデータの保存に適している．「IMF02MA1」は2MBのデータ保存が可能なフラッシュメモリを用いており，書き込み後のデータ保存のためのバックアップ電源が不要．両モデルとも専用メモリコントローラをを搭載し，外部からのノイズ，静電気，挿抜時の誤動作を防ぐ．価格は，1MBタイプの「IMS01MA」が3万2千円，2MBタイプの「IMF02MA1」が3万円．

MicroDevice

AMD，単一5V電源4Mビットフラッシュメモリを発表

日本AMD㈱(03-3346-7560)は，単一5V電源，セクタ消去機能付き，4Mビットフラッシュメモリ Am29F040を発表した．同製品はEmbedded ProgramとEmbedded Eraseにより最低10万回/セクタを保証し，書き込み/消去をチップ上のアルゴリズムで自動的に行うため，書き込み時のCPUの負荷を免除する．



Am29F040のパッケージには32ピンのPLCC, TSOP, LCC, PDIP, およびDIPがある．価格は32ピンPLCC，90nsが3,800円，32ピンTSOPパッケージが4千円(いずれも千個/ロット時)．

Hardware News

Hardware News

DELL，VL-Bus搭載ハイエンドモデル

デルコンピュータ㈱(0120-1984-86)は，デスクトップ型パソコンの新シリーズとしてVL-Busスロットを搭載し，グラフィック性能も強化した「OptiPlex 486」シリーズ2機種6タイプを発売する．同シリーズは「486/L」,「486/M」シリーズの後継機種にあたり，スリムなシャーシの「OptiPlex 486/LV」シリ

TB TIME

ーズと，スロットとドライブベイを増やした「OptiPlex 486/MXV」シリーズの2機種からなる．それぞれ，CPUにi486DX2 /66MHz，i486DX/33MHz，i486SX/ 25MHzを搭載した3タイプのモデルが用意される．価格は，「OptiPlex 486/VL」が19万5千円～，「OptiPlex 486/MXV」が23万8千円～．



Hardware News

キヤノン，DOS/Vパソコン 3機種9モデル発売

キヤノン販売㈱(03-3455-9127)は，DOS/VをサポートするIBM PC/AT互換機「INNOVA」シリーズを発売する．同シリーズは，CPUにi486SX/33MHz(4233SX)，i486DX/33MHz (4233DXC)，i486DX2/66MHz (4266 DX2)を搭載した3機種からなり，メモリは標準8MB(FDモデルは4MB)，3モード対応FDDを装備．表示部にはCIRRUS LOGIC社のCL-GD5428を搭載．拡張スロットはISA×4(うちVL-Bus×2)となっている．価格は，4233SXのFDモデルが19万2千円，同じく4233DXCのFDモデルが22万4千円，4266DX2のFDモデルが27万2千円．



Hardware News

東芝アメリカ情報システム，新型ノートブック型パソコン

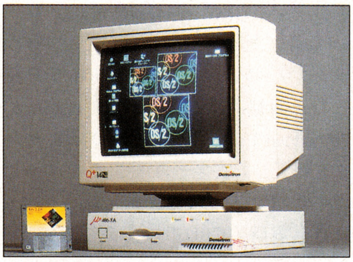
東芝アメリカ情報システム社は，ノートブック形パソコン「T4700C」シリーズ4機種を発売した．同製品は，CPUに3.3V SLエンハンス i486DX2/50MHzを搭載．ウエーブ・オーディオ対応，グラフィックアクセラレータとローカルバス技術を使用することによって動画にも対応，PCMICA 2.01準拠ICカードスロット，新型ボールポイントマウス2.0ポインティングデバイス装備，などの特徴を持つ．同シリーズには，9.5インチSTNカラー液晶ディスプレイを搭載した「T4700CS」，9.5インチSVGAカラーTFTアクティブマトリクス液晶ディスプレイを搭載した「T4700CT」の2タイプがあり，それぞれ200MB/300MBのHDDモデルが用意される．価格は「T4700CS/200」が4,599ドル，「T4700CS/320」が4,999ドル，「T4700CT/200」が5,299ドル，「T4700CT/320」が5,699ドル．



Hardware News

デンシトロン，OS/2 J2.1 プリインストールモデル

㈱デンシトロン(03-3767-9701)は，日本アイ・ビー・エム㈱と，IBM OS/2 J2.1のプリインストール契約を締結．同社のLAN機能標準搭載のデスクトップPC「μ+486-XA」にOS/2をプリインストールして発売する．同モデルは，Novell社NE2100互換のAMD社製LANチップをマザーボード上に搭載，10 BASE-Tコネクタを背面パネルに標準装備しているため，コネクタを差し込むだけで，そのままイーサネットに接続できる．ビデオシステムにはIBM XGA-2を採用．価格は28万円から44万円．



Hardware News

エプソン，UG NOTE PC-486NOTE ASD

セイコーエプソン㈱(03-3377-3531)は，「エプソンUp Grade NOTE PC-486NOTE ASシリーズ」の新モデルとして，「エプソンUp Grade NOTE PC-486NOTE AS D(PC486NASD2)」をエプソン販売㈱(03-3377-7001)を通じて発売する．

同製品はCPUにSLエンハンスi486DX2/50MHzを採用し，メモリ5.6MB，TFTカラーLCD(640×480ドット)を搭載する．また，「PC-486NOTE ASシリーズ」および「PC-386NOTE AR」のオプションとして，SLエンハンスi486DX2/50MHzを搭載した「アップグレードCPUモジュール(PCCPUM22)」も発売される．価格は「PC486NASD2」が54万8千円，「PCCPUM22」が10万5千円．

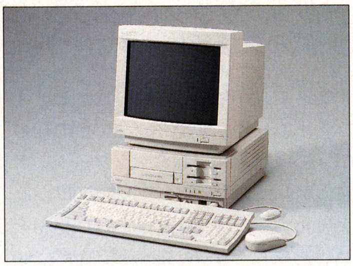
Hardware News

プロサイド，10万円ぽっきりコンピュータ

プロサイド㈱(03-3255-1517)は，AT互換機2機種も同時に発売する．「10万円ぽっきりコンピュータ JD1991DX-401X」はCPUにi386DX(40MHz)を採用し，メインメモリ4MB，ハードディスク120MB，3.5インチ3モードFDDを搭載する．価格は10万円．「20万円コックリコンピュータ JD1994DX2-50MVX」はVL-Bus対応で，CPUにi486DX2(50MHz)を採用．またメインメモリ4MB，ハードディスク200MB，3.5インチ3モードFDDを搭載する．価格は20万円．

Hardware News

富士通，FM TOWNSⅡの ラインアップを一新



富士通㈱(03-5690-0032)は，マルチメディアパソコン「FM TOWNSⅡ」の新機種および周辺機器の大幅な強化を行い，順次発売する．FM TOWNSⅡの新機種には，10万円台のエントリーモデル「モデルME」(i486SX/25MHz)，高解像度表示を標準でサポートする「モデルMA(i486SX/33MHz)/MX(i486DX2/66MHz)」の3機種8タイプを投入．

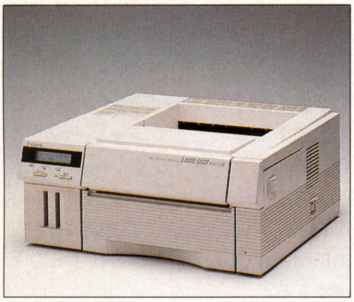
周辺機器としては，4モード(15/31/48/56KHz)マルチスキャン対応14インチディスプレイ「FMTDP8311」．24bitカラーのビデオキャプチャー機能・Video in Windows機能を持つ，モデルMX/MA専用ビデオカード「ビデオカードⅢ」．モデルME/MAをそれぞれ，i486DX2/50MHz，i486DX2/66MHz相当にアップグレードする「オーバードライブカード」などがある．価格は以下の通り．

FM TOWNSⅡ ME20	(メモリ2MB,2FDD)	19万8千円
ME170	(メモリ2MB,170MBHDD)	26万8千円
FM TOWNSⅡ MA20	(メモリ4MB,2FDD)	29万8千円
MA170	(メモリ4MB,170MBHDD)	36万8千円
MA340	(メモリ4MB,340MBHDD)	41万8千円
FM TOWNSⅡ MX20	(メモリ4MB,2FDD)	39万8千円
MX170	(メモリ4MB,170MBHDD)	46万8千円
MX340	(メモリ4MB,340MBHDD)	51万8千円
14インチカラーCRTディスプレイ		11万5千円
ビデオカードⅢ		5万9,800円
オーバードライブカード	(モデルME用)	10万円
オーバードライブカード	(モデルMA用)	12万円

Hardware News

キヤノン，レーザショットに 省電力タイプの新機種

キヤノン㈱(03-3455-9320)は，A4対応プリンタ「レーザショット LBP-A404GII」を発売する．同プリンタは解像度600dpiでありながら画像処理技術「スーパー・スムージング・テクノロジー」によって横1,200dpi×縦600dpi相当の印字を実現する．また「オンデマンド定着方式」の採用により従来の5分の1の消費電力となっている．価格は14万8千円．



Hardware News

カシオ，高速処理A3対応 ページプレストCP-3250

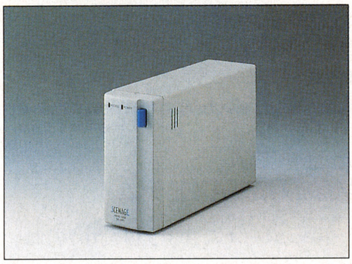
カシオ電子工業㈱(03-5350-0761)は，32ビットRISCプロセッサを搭載し，グラフィックデータやイメージデータの高速処理を実現したA3判対応ページプリンタ「ページプレスト CP-3250」を発売する．400dpiの基本解像度に独自のスムージング機能を加えることにより，600dpi相当の高品位な印刷ができる．価格は39万8千円．

Hardware News

ICM，SCSI-2対応HDD SCENAGEシリーズ

㈱アイシーエム(06-648-4702)は，PC-9800シリーズ用SCSI-2対応ハードディスク「SCENAGE(セナージュ)シリーズ」を発売する．同シリーズには，バスマスタ方式のSCSI-2対応のインターフェースが付属したDBモデル，SCSI専用スロット対応のインターフェースが付属したFBモデル，そしてインターフェースが付属しないBLモデルがある．価格は，インターフェースが付属するDB，FBモデルで，容量約126MBのSG-120が6万4,800円，179MBのSG-180が7万4,800円，256MBのSG-270が8万4,800円，約325MBのSG-340が9万9,800円，約513MBのSG-540が13万8千円となっている．インターフェースが付属しないBLモデルでは各1万円安．

なお，同社はDOS/Vマシン対応版の「SCENAGE SGシリーズ」も発売した．



Hardware News

緑電子，新98MATE/FELLOW ファイルベイ用HDD

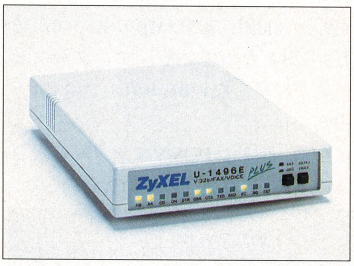
緑電子㈱(044-989-7633)は98FELLOW，98MATEの新シリーズに採用されたファイルベイに対応した内蔵ハードディスクユニット「FBシリーズ」を発売する．価格は以下の通り．

FB240	240MB	4万4,800円
FB340	340MB	6万4,800円
FB510	510MB	10万3千円

Hardware News

ZyXELモデム 国内仕様モデル発売

㈱エミックスエンジニアリング(03-5213-8444)は，ZyXEL(ザイセル)社のモデム「U-1496E PLUS」日本仕様モデルの電気通信機器審査協会の技術基準適合の認定を受け，11月25日に販売開始した．同製品はV.32bisのほか，独自モードの16.8kbps，19.2kbpsの転送スピードを持つ．またMNP 3-5，V.42/V.42bisにも対応．FAXはV.17に対応し，EIA Class 2 FAXコマンド，ZyXEL FAX ATコマンドセットを持つ．価格は8万6千円．



■ZyXEL U-1496E PLUS

Hardware News

ヒューコム, V.fast対応FAXモデム

㈱ヒューコム(03-5397-7077)は，V.fast対応のFAXモデム「The V.fast」の販売を開始する．同モデムは米Microcom社の製品で，通信速度はV.fastで最大28,800bpsをサポートし，またV.32bis/V.42bis/MNP10/MNP5/EIA-578(Class 1)をサポートしている．また同モデムではパラレルポートインターフェースを内蔵し，AT互換機で端末速度最大115.2Kbpsを実現する．価格は19万8千円．



Hardware News

Hayes, V.fast対応モデムをアメリカで発売

米Hayes社は，アメリカおよびカナダにおいてV.fast対応の「OPTIMA288 V.FC+FAX」を発売した．同モデムはV.fast対応で通信速度は最大28,800bps．またV.32bis/V.42bis/V.42/EIA-578(Class 1)をサポートする．価格は579ドル(アメリカ)，699ドル(カナダ)．

Hardware News

メルコ，サウンド+グラフィックアクセラレータボード

㈱メルコ(052-619-1827)は，ビデオコントローラにCIRRUS LOGIC社のCL-GD5428を採用したグラフィックアクセラレータボード「WAB-1000/2000」を発売する．「WAB-2000」はDRAMを2MB搭載し，MS-Windowsで最大1,280×1,024ドット/256色(インターレース時)，1,024×768ドット/6万色(ノンインターレース)の表示が可能．「WAB -1000」はDRAMを1MB搭載で，1,024 ×768ドット/256色(ノンインターレース)が可能．また両機種ともメルコ独自のCD-ROM拡張コネクタを装備．これらのボードに接続する倍速CD-ROMドライブの発売も予定している．価格はWAB-2000が2万9,800円，WAB-1000が1万9,800円．

またグラフィックアクセラレータ機能に加え，PCMサウンド機能(YAMAHA YMZ263)を同一ボード上で実現した



「WSR-E」も同時に発売する．グラフィックアクセラレータ部の諸元はWAB-1000と同じ．価格は2万9,800円．

また98FELLOW/MATE用に，内部増設RAMボードEAB-4000T(2万9,800円)，グラフィックアクセラレータボードWAB-S(2万9,800円)，Micorsoft Windows 3.1をセットにした「WBX-P」を限定3,000台で発売する．価格は5万4,800円．

Hardware News

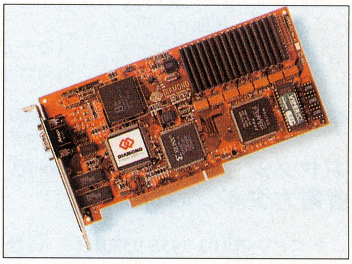
ソフトウェアジャパン, BOCA社FAXモデムを発売

㈱ソフトウェアジャパン(03-3862-2754)は，ボカリサーチ社の高速FAXモデム「M144JE」を発売する．同製品はデータ部はV.32bis/V.42bisに，FAX部はV.17/EIA-578,592(Class 1,2)に対応している．また，ファックス・データ通信ソフトとしてスミスマイクロソフトウェア社の日本語版「Quick Link II Fax for Windows」が添付される．価格は3万5千円．同時に，IBM PC/ATコンパチ用高速シリアルカード「IOAT55」も1万5千円で発売される．

Hardware News

DIAMOND Computer, VIPER PCIなど

㈱ダイヤモンド・コンピュータ・システムズ・ジャパン(03-5489-7330)は，ビデオキャプチャ・圧縮・再生カード「VideoStar/Pro」，サウンドカード「SonicSound LX/DX」，グラフィックアクセラレータカード「Stealth 32 PCI/VLB」，「Viper PCI」を発売する．



■VIPER PCI

「VideoStar」はリアルタイムのWindows上での動画表示機能に加え，フルモーションビデオキャプチャ，圧縮，再生機能をサポートしたカードで，320×240の解像度で30フレーム/秒の速度で動画を記録することができる．また，「VideoStar Pro」ではハードウェアにより，動画のJPEG圧縮が可能．

「SonicSound LX」はヤマハのOPL-3を搭載，ステレオFM音源を20個サポートし，さらに16bit PCMの録音，再生機能を持つ．「SonicSound DX」では上記の機能に加え，ウェイブテーブル音源を24個サポートする．両製品ともSound Blaster，Suond Blaster Pro，Windows Sound Systemと互換性を持ち，IDE CD-ROMドライブを接続することができる．

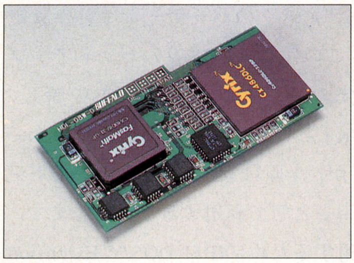
「Stealth 32」はVLバス，PCIバスをサポートし，2MBのDRAMを搭載している．グラフィックスコントローラにはTseng LabのET-4000/W32pを搭載し，最高で1280×1024ドット256色，1024×768ドット65,536色，800×600ドット/640×480ドット1,670万色の表示が可能．「Viper PCI」は「Viper VLB」のPCI版で，WeitekのPower 9000を搭載している．全製品日本での価格は未定．

Hardware News

メルコ，286用倍速CPUアクセラレータ

㈱メルコ（052-619-1827）は，Cx486DLCを搭載した286用CPUアクセラレータボード「HRX-12T/W」を発売する．

「HRX-12T」はPC-9801DX/RX/EX対応で，CPUクロックを本体クロックの3倍速の36MHzで動作させる．また「HRX-12W」は本体クロックの2倍でCPUを動作させる．価格はHRX-12Tが2万4,800円，HRX-12Wが1万9,800円．また数値演算コプロセッサとのセットも同時に発売する．

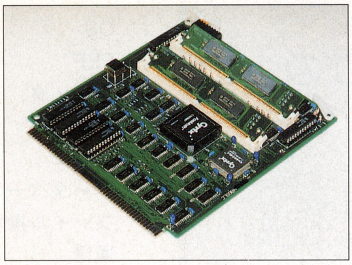


■HRX-12W

Hardware News

データシステムプロジェクト98用CPUアクセラレータ

㈱デーシステムプロジェクト（0776-23-5353）は，Cx486SLC-33を搭載したPC-9801用CPUアクセラレータボード「Dpro486」を発売した．同製品はCPUに8086またはV30のみを搭載した旧型の機種用で，拡張スロットに挿入する方式のため本体に改造を必要としない，数値演算コプロセッサとオンボードメモリをオプションで準備しており拡張が可能，などの特徴を持つ．価格は5万円から．



■Dpro486

Hardware News

マクサス，i386SXからIBM 486SLC2への換装サービス

㈱マクサス インターナショナル テクニカルセクション（03-5940-2811）では，i386SXをIBM486SLC2に換装するサービスを開始した．IBM486SLC2は（1）16KBの内部キャッシュ，（2）クロックダブラにより内部倍速動作，（3）ハードウェアバススヌーピング，（4）3.3V動作，（5）Intelオリジナルのマイクロコード使用によりi486完全互換などの特徴がある．価格はi386SXからの換装が5万9,800円，マクサスでCx486SLCに換装済なら4万9,800円，他社でCx486SLCに換装済なら5万4,800円となっている．

Software News

Software News

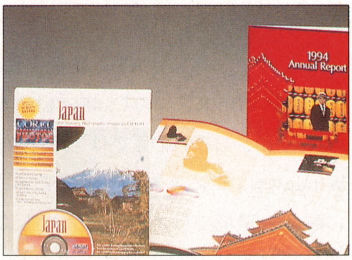
マイクロソフト，Access for Windows 発売

マイクロソフト㈱（03-5454-2300）は，リレーショナルデータベース管理システム「Microsoft Access for Windows Version1.1日本語版」（以下Access 1.1）を発売する．同製品は独自の学習ツールを装備するなど，初心者にも簡単に扱えるように考慮されており，また，一連の作業を自動化できるマクロ機能，高機能なアプリケーションを設計するためのAccess Basicなどの機能も持っている．さらに，Microsoft Exel，Lotus 1-2-3のデータが利用できるほか，桐V4やdBASEなどの既存のデータベースファイルをそのままAccess 1.1のデータとして利用する事も可能．ODBC経由でのさまざまなデータベースとの接続，セキュリティ機能の充実も行われている．標準価格は5万8千円だが，発売日から1994年2月28日（月）まで，特別価格2万5千円で販売する「発売記念キャンペーン」を実施する．

Software News

日本語Windows3.1対応 COREL DRAW 3.0/J発売

住友金属工業㈱（03-5476-9816）は，カナダCOREL社のグラフィックソフト「COREL DRAW 3.0/J」の日本語版を発売する．同製品は，ロールアップウィンドウを使ったマウス操作により，オブジェクトの回転・傾斜・立体化・ブレンド・ミラー化など，思い通りのデザイン作成・加工が簡単に実行できるCorelDRAW，スキャナから取り込んだ写真・絵などのレタッチができるCorelPHOTO-PAINT，スキャナで読み込んだビットマップ画像のトレースを行えるCorelTRACEなど5種類のソフトが含まれる．価格は5万8千円．



■COREL-PHOTO

Software News

プロサイド，DOS/Vゲームに進出

プロサイド㈱（03-3255-1517）は，台湾PANDA Entertaiment Technology社と提携し，DOS/Vアクションゲーム「三國志武将争覇」を発売する．価格は5,900円．同時にオリジナルゲームパッド（1,900円）も発売する．



元98ユーザによる

98ユーザのための AT互換機入門

●著者●
佐伯英和
Hidekazu SAEKI

98環境の移植

前回は簡単なマシンの紹介に終わりましたが，今回からはいよいよ実際の使用を前提としたレポートをしていきましょう．

ところで，筆者がAT互換機を使用する以前は，PC-9801DS（VIPERによりCx486SLC/16MHz化）を使用していました．DOS環境では，それなりに快適でしたがWindowsはインストールしてあるだけで，とても使用に耐える環境ではありませんでした．しかし，AT互換機を手に入れたからには，コンピューティング環境をWindows上に移行させようと考えています．

そこで，いままで98で使用していた環境をなるべく変更しないようにしながら，ATの環境設定を行っていきましょう．まず，いままで98で主に使用していた環境を挙げておきましょう．

OS	MS-DOS Ver.3.3
FEP	WXⅡ$^+$
エディタ	VZ Editer
ワープロ	松 Ver.6
表計算	アシストカルク
データベース	アシストカード
ファイル操作	FILMTN（フリーソフト）
通信	CCT98-Ⅲ

と，こんなところでしょうか．このうち，アプリケーション関連は別として，FEPのWXⅡ$^+$と，エディタのVZ Editer，通信のCCT98-ⅢはAT互換機に移行したとしても手放せません．幸い，これらのソフトはDOS/V上で動作するものが手に入りますので，同一環境での使用が可能です．その他のものは，せっかくですからWindows上で動作するアプリケーションを使用することにします．

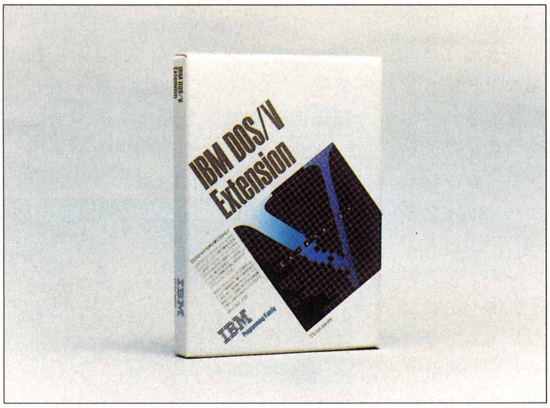


写真1●IBM DOS/V Extension

DOS環境の整備

まず，何はともあれIBM DOS J5.02/Vをインストールする必要があります．と言っても，ブートディスク（DISK1）をセットしてセットアップを実行します．あとは画面の指示に従うだけでOKです．一つだけ注意するとすれば，ハードディスクのパーテーションの設定ぐらいでしょうか．この作業はDOS/Vのセットアップでは省かれていますので，セットアップを中断し，DOSに下りて「FDISK」というコマンドを実行しなくてはなりません．ですが，今回は340Mバイトのハードディスクを1パーティションとして利用することにします．

次に，V-textを利用するためにIBM DOS/V Extension（**写真1**）をインストールします．これも，指示に従えば無事インストール完了です．あとはVZ Editor，CCT/V，各種ツールをインストールしてDOS環境の整備は完了です．

ところが，いざ使用してみるとVZ Editorの調子がよくありません．画面にゴミがたくさん出てしまい，とても見づらいのです．いろいろ調査した結果，どうやらDOS/V Extensionとの相性が良くないようです．しかも，このマシンでのみ発生します．自宅にあるもう1台のAT互換機では，このような現象は起こらないのです．ビデオカードとDOS/V Extensionとの相性の問題のようです．まあ，AT互換機を使用する上でなんらかのトラブルが発生するのは覚悟していましたが，こんなに早くトラブるとは，これから先は…．いや，考えるのは

*Profile:*佐伯英和 '71年生まれ．某体育会系ライターの弟子で，ライター家業は新人．仕事はパソコン．趣味は競馬の不良学生．

やめましょう．

で，しばらくは我慢して使っていたのですが，DOS/V Extensionの最新版（1.01）が手に入ったので，これを何気なくインストールしたところ，あっさり解決してしまいました．ちなみに，1.00から1.01へのPTFファイルは，NIFTY-Serve（FIBMPRO lib10 #68,69）などで手に入れることができます．この1.01では，以前はサポートしていなかったS3社のグラフィックアクセラレータチップを搭載したビデオカードに正式対応し，インストーラがシステム構成を判別して，S3のグラフィックアクセラレータチップが使用されていれば，インストーラからS3用ドライバを選択できます．ですから，以前のように別途S3用のドライバーを購入する必要はありません．これで，VZ EditorはV-textの106桁×39行という広いDOS画面（**写真2**）での超快適な使用が可能となりました．

Windows環境の整備

さて，Windowsですが，今回はIBM版のものを使用しました．これは，現時点ではMS版のWindowsよりも安定して動作するからです．特にDOS窓関連は，MS版よりも安定しています．インストール作業も，これまた簡単にできました．指示に従ってディスクの入れ替えをおこなえばOKです．まずは，デフォルトの設定のままインストールしてやって，細かな設定はあとから行うことにしましょう．

後から行う設定といっても，ディスプレイ（表示解像度）の設定ぐらいです．このマシンには日本語Windows用のディスプレイドライバが標準で添付されていますので，これを使用すれば最大で1,280×1,024ドット/16色の解像度でWindowsを使用できるようになります．Windowsをインストールしたら，DOSからWindowsのディレクトリに移動して「Setup」コマンドを実行．ディスプレイの項目を変更して，1,024×768ドット/256色の解像度で使用することにします．

今回使用しているMiTAC VL-STATiONに搭載されているビデオカード（**写真3**）には，VRAMが1Mバイトしか搭載されていませんので，色数は1,280×1,024ドットで16色，1,024×768ドットで256色までしか使用できません．解像度を落とせば，6万色まで使用できますが，狭い画面ではWindowsを使用するメリットがないのと，このビデオカードで使用されているS3の86C805では256色以上の色数を要求するとスピード的にも問題が生

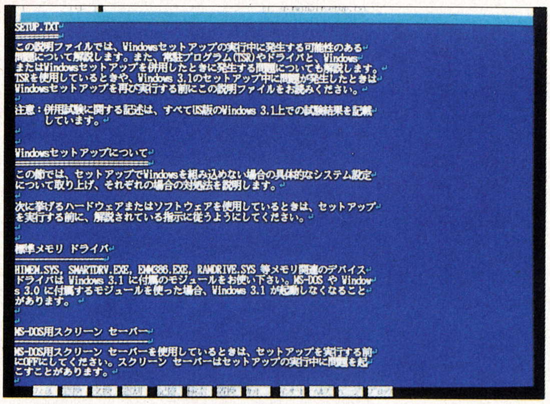
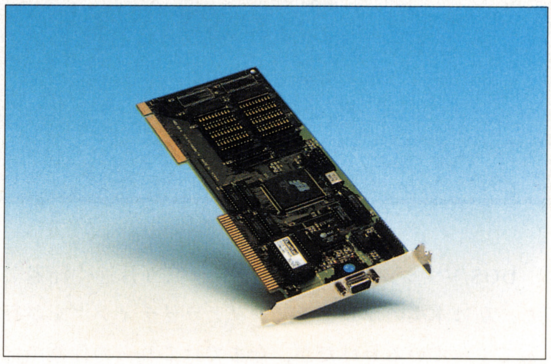


写真2●VZ EditorのV-text画面


写真3●86C805を搭載した標準装備のビデオカード

じるようなので，1,024×768ドット/256色という選択に至りました．

ここで注意すべきは，Windows環境の変更を行う場合には，CONFIG.SYSとAUTOEXEC.BATを変更して，余計な常駐プログラムを削除しておくとよいということです．今回も，マニュアルに記載されている通りに，DOS/V ExtensionをCONFIG.SYSとAUTOEXEC.BATから削除してからセットアップを行いました．Windows環境の変更時には常駐プログラムはトラブルの原因になりかねませんから注意が必要です．

次回に向けて

これで，DOS，Windowsともに使用できる環境が完成しました．トラブルらしいトラブルはDOS/V Extensionだけでした．安定して動作するってぇのは，いいことです．そう考えると98はいいマシンです．なにも考えなくても安定してますから．次回もノントラブルだといいなぁと祈りつつ次号までしばし休息．

製品名：	MiTAC VL-STATiON
価格：	43万8千円
問い合わせ先：	㈱マイタックジャパン
	☎03-5420-2822

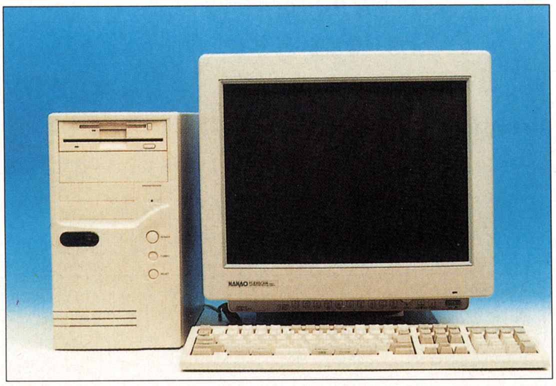
プログラマによる

プログラミングのための AT互換機入門

●著者●
広野忠敏
Tadatoshi HIRONO

DOS/Vマシンを使い始めてもう二カ月近く経過した．私は本業のプログラミングとライティングを並行してやっているわけだが，今まではこれらの作業のほぼすべてを98を使ってやっていた．Fujiyama（**写真1**）が来てからは，これらの作業をDOS/Vマシンでやるべく環境を構築したわけだが，なかなかスムーズに環境を移行することができた．今となっては，98はほとんど使っていないという状況になってしまった．さすがに，ターゲットが98での開発は98でやるのが手っとり早いが，ライティング作業やその他の雑用，ターゲットが98以外での開発のほぼすべてをWindows上でやっているというのが現状だ．

さて，二カ月間DOS/Vマシンをいじり倒しているわけだが（ほぼ24時間稼動し続けている），実はこれといったトラブルはまだ経験していない（さっぱりなにも拡張していないからだという話はあるが…）．当初は，DOS/Vは98に比べてハードウェアの面でも，ソフトウェアの面でもトラブルが多いという話を聞いていたが，このマシン（Fujiyama）は極めて安定して動作しているため，現状ではまだハードウェア的なトラブルは発生していない．


写真1●Fujiyama

AT互換機は，98などの国産機にくらべてハードウェアの面でも，ソフトウェアの面でも柔軟に造られている．サウンドカードやSCSIインターフェースなどサードパーティー製のデバイスも多い．さらに，同じ互換機といえども，マザーボードなどの設計は微妙に違うことも多い．もちろんBIOSのコードもメーカーによって微妙に異なる．AT互換機のトラブルは，たとえばマザーボードと各種のインターフェースの相性が悪かったり，割り込みの設定や各種コンフィグレーションなどの設定が誤っているケースで発生することが多いようだ．したがって，初心者には使いにくいマシンであるという話を聞くが，正直いってそんなことはないと思う．

ハードウェアのトラブルや相性はAT互換機に限らず，98などの国産マシンでも存在するし，もちろん設定などのユーザに起因するミスも同じ様に発生する．AT互換機が98などに比べて難しい印象を与えるのは，単にAT互換機についての正しい情報が手に入れにくいからではないだろうか．さすがに，最新の情報を常に入手するのは難しいが，AT互換機専門誌やパソコン通信など，昔にくらべて情報は比較的入手しやすくなっている．また，ショップの情報も貴重だ．これらの情報をうまく活用すれば，AT互換機なんてちっとも難しくないだろう．

Microsoft Visual C++

前号でも書いたが，AT互換機は極めてプログラミングに適したマシンだ．先月はBorland C++（以下BC++）をインストールして，Windowsのプログラムを色々と作ってみたが．今回はMicrosoft Visual C++（**写真2**）を使う機会があったので，簡単に紹介しよう．

Windowsでプログラムを開発するには，Microsoft C/C++ Ver7.0（以下VC++）などのコンパイラとSDK（Software Developers Kit）を使う方法．BC++などのWindows上の統合開発環境を使う方法．Visual BASICなどのよりお手軽な開発環境を使う方法がある．VC++はBC++と同様に，Windows上に統合開発環境を

Profile:広野忠敏 '62年生まれ．流しのプログラマ兼ライタ．3.0発売当時からWindowsプログラミングに手を染めるが開発マシンがPoorで苦労する．

構築するコンパイラだ．

VC++では，Microsoft C/C++ Ver7.0でサポートされたMicrosoft Foundation Class（MFC）と呼ばれるクラスライブラリを搭載して，このライブラリを使うことによって効率良くWindowsのプログラミングをすることができる．BC++のOWLやVC++のMFCはWindowsの低レベルなAPIをカプセル化したクラスライブラリというわけだ．ご存知のように，Windowsのプログラムはすべてメッセージを基準に動作している．たとえば，ウィンドウクライアントでキーが押されたらそのウィンドウにキーが押されたというメッセージが送信される．また，ウィンドウが移動するときなども同様にメッセージが送信される．このようにメッセージが基本となっているシステムであるため，C++などのオブジェクト指向言語とは相性がいい．

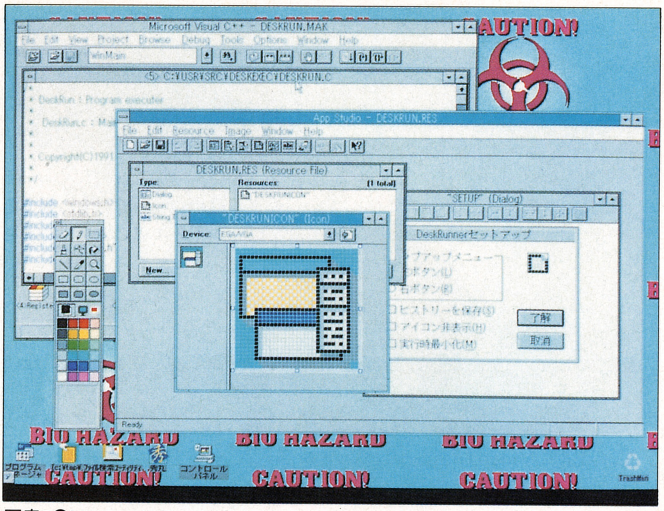


写真2●Microsoft Visual C++

さて，VC++はDOSベースのコンパイラやユーティリティ，Visual C++，AppStudioの三つで構成される．Visual C++がVC++の中核をなす部分で，これがWindowsでの統合開発環境となる．ここから，コンパイラやリンカなどを呼び出して，プログラムのコンパイルやリンクが行われる．統合開発環境の機能には，エディタやデバッガ，クラスブラウザの他にも，AppWizardとClassWizardと呼ばれるものがある．AppWizardはいわゆるプログラムビルダーで，これを使えばWindowsプログラムのプロトタイプとなるコードを生成してくれる．ただし，ここで生成されるコードは，MDIやOLE，さらにはツールバーなどの機能を持ったかなり高度なものだ．AppWizardで生成されたコードに肉付けをするというスタイルでよりお手軽にWindowsアプリケーションを作ることができる．

ClassWizardは文字通りクラスを管理するためのツールだ．ClassWizardでは，AppWizardで作られたソースのクラスとウィンドウメッセージ，メンバ関数をハンドリングする．たとえば，あるアプリケーションのウィンドウでマウスが押されたら‥‥するというコードを書くには，ClassWizardでクラスとオブジェクトを選択して，さらにWM_MOSEMOVEメッセージを選択，そして関数に目的のコードを書くというように極めて簡単にそれぞれのメッセージの処理を実装することができる．

AppStudioはプログラムのリソースを作成するツールだ．AppStudioを使えば，ダイアログやビットマップ，メニューリソースなどを簡単に作ることができる．さらに，VBXコントロールを使うことも可能だ．また，EXEやDLLからリソースを抜き出すこともできる．SDKなどで開発しているとリソースを作るのは，けっこうたいへんな作業になるが，AppStudioやBC++のWorkShopなどのツールがあれば，リソースを作るのもそんなに苦にはならないだろう．VC++にはまだまだ色々な便利な機能があるので，機会があれば改めて紹介したい．

さて，BC++にVC++とFujiyamaにもやっとまともな開発環境がそろってきた．ちなみに，BC++とVC++両方あわせて約100Mバイトのハードディスク領域を消費している．一昔前なら，コンパイラ環境だけで100Mバイトも使うなんて冗談みたいな話だったが，これが現実だ．アプリケーションと共に，開発環境やプラットホームもどんどん肥大化する傾向にある．汎用機などのメインフレームの世界では，ダウンサイズと称して環境や資源を縮小する傾向にあるが，パソコンの世界では逆のようだ．早くも340Mバイトのハードディスクの残りは90Mバイト弱である．

製品名：	Fujiyama
価格：	22万8千円
問い合わせ先：	くまくま
	☎0975-46-3720
	14:00〜19:00（土日祝日定休）
	FAX 0975-43-8451 24時間受付

Light Rave

最近，AMIGAユーザたちの間で話題の絶えない商品がある．それがLight Rave(以下，LR)．
「あれ どこかで聞いたことのあるような名前だぞ」と思った方はかなりスルドイ．
今回は，久々に現れたAMIGAらしいプロダクツ“LR”をみていこう．

●著者
桃園隆夫
Takao MOMOZONO

このLR，VideoToasterにバンドルされてくる3Dレンダリング&アニメーションパッケージ“LightWave3D”(以下，LightWave)と名前が似ていると思わないだろうか? そう，実はLRはLightWaveを実行させるためのモジュールだ．しかもLRを使うと，VideoToasterボードがなくてもLightWaveが使えるようになってしまう．Toasterボードなしで LightWaveが使えるということは，ビデオスロットなしのAMIGAでも“憧れの”(?)LightWaveが実行できるわけだ．

ちょっと前に「こんな商品はないかな?」と言っていたものが，ほとんど確実に現実になる．それもAMIGAの世界の摩訶不思議なところだ．

LIGHT RAVE

LRは，ソフトとハードをペアで使用するパッケージになっている．もちろん，このパッケージにはLightWaveとModelerのプログラムは含まれていないので，LRとは別にそれら二つのプログラムを用意する必要がある．LRで起動可能なLightWave/Modelerのバージョンは2.0または3.0．3.0を使う場合にはModelerで必要なファイル(マクロファイルなど)も用意しておいた方がいいだろう．

LRを起動するにはまずLRのメインプログラムをハードディスクへインストールし，同じハードディスク上にLightWave/Modelerをコピーする．LightWave/ModelerはAMIGAシステムに搭載されているCPUによってノーマルまたは“FP”のどちらか一方が実行されるので，CPUが68000のユーザは通常のLightWave/Modelerを，680xxの場合は“FP”の方だけを使用すればOKだ．

LIGHT RAVE HARDWARE

一方，LRのハードの方は，AMIGAのシリアルポートに差し込む．ハードといっても大きさはタバコの箱を半分にしたほどのものだが，これがLightWave/Modeler両者を騙し(?)，あたかもAMIGAにVideoToasterボードが付加されているかのように見せかける大胆なヤツなのだ(**写真1**)．

ところで，シリアルポートにLRのハードを差してしまうと，モデムやプリンタなどのシリアルを使うデバイスが使えなくなってしまう．通常，LRとそれらのデバイスを同時に使用することはまずないと思うが，シングルフレームコントローラのように，LightWaveと同時に使うデバイスもある．シングルフレームコントローラは，LightWaveなどでレンダリングした絵をVTRへ1フレームずつコマ撮りするときに使うデバイスで，

Multimedia PC from GAME up to UNIX

AMIGA

これが使えないとなると少々困るユーザも多いはずだ．

シングルフレームコントローラを使う場合，LightWaveでレンダリングしたフレームデータをいったんHD(ハードディスク)へセーブしておき，あとでコマ撮りする方法もある．が，実はLRでいったんLightWave/Modelerを実行させてしまえば，AMIGAの電源を切らない限りLightWave/Modelerは再実行ができる．つまり，最初にそれらのプログラムを実行させるときにだけLRのハードをAMIGAのシリアルポートに差しておけば，ハードを抜いても，何度でも再実行可能になっているわけだ．

どうしてもLRとシングルフレームコントローラを同時に使いたい場合は“シリアルスイッチボックス”を使う方法もある．このボックスは名前の通り，シリアルポートをスイッチでいくつかに分配するもので，一つにLRのハードを，もう一つにシングルフレームコントローラを差し，スイッチでそれらを切り替える．これを使ってLR+LightWave/Modelerとシングルフレームコントローラとを同時に使えることは確認ずみだが，**これによって発生したマシンのトラブルに関しては一切保証ができないので，シリアルスイッチボックスを使う場合は，各ユーザの責任によって行ってください．**

また，5～8個のシリアルポートが1枚のペリフェラルに搭載されている，いわゆる「デュアルシリアルポート」でも試してみたが，今のところシングルフレームコントローラはまだ作動していない．これに関しては，各ソフトが対応になるまで待つしかなさそうだ．

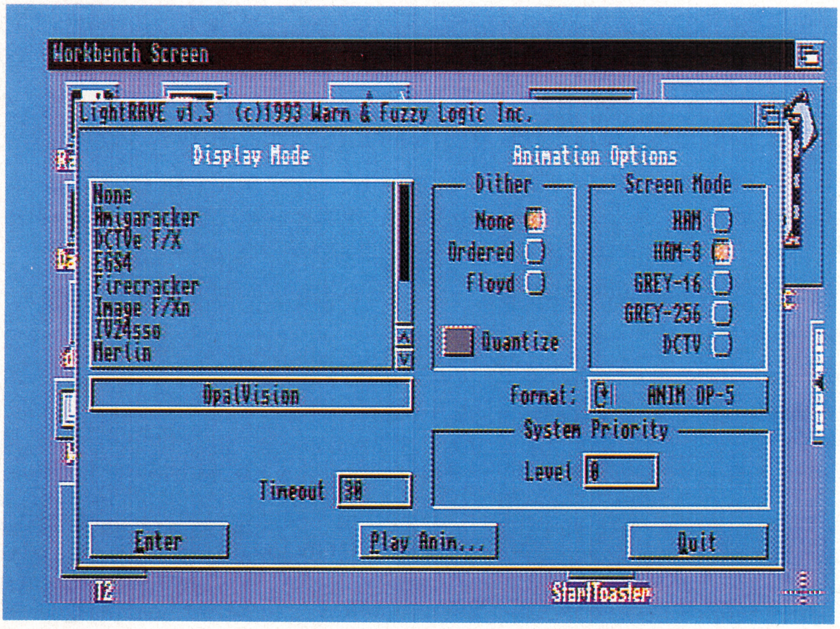
写真1●LightRave本体



LIGHT RAVE SOFTWARE

LightWave/Modelerを動作させるには，まずLRのソフトを実行する(**写真2**)．

写真2●Light Raveのオペレーティングソフトの画面



AMIGA
Light Rave

写真3●LightWaveの画面

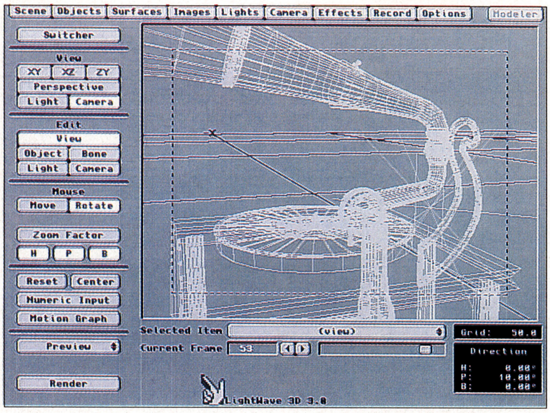
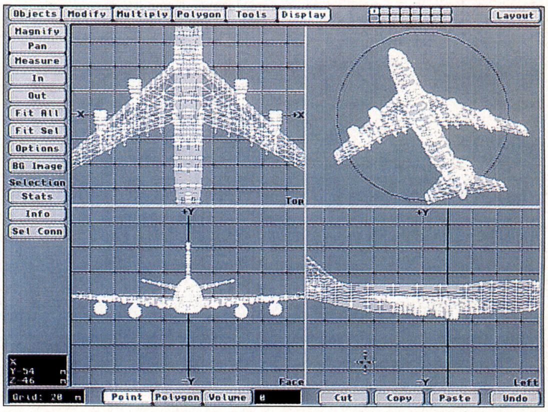


写真4●Modelerの画面



LRのソフトでは表示とアニメーションのモードを設定した後に“ENTER”コマンドをマウスでクリックするとLightWaveが実行される．LRがサポートする画面表示モードは，AMIGA(HAM)をはじめ，“Firecracker”“DCTV”“OPALVISION”“IV 24”“EGS”“Merlin”の6種のフレームバッファ，そしてGVPのプロダクツ“Image F/X”だ．

LightWaveは通常(つまりVideoToasterを使うと)，レンダリングされた絵はToasterのフレームバッファ(DV1)へ送られ フルカラーで表示される．そしてLRでは前述したフレームバッファをDV1の代わりに使う．

これまで，LightWaveは使いたかったけれど，Toasterはチョットおおげさすぎると思っていたユーザやOPALVISION，あるいはFirecrackerユーザもやっとLightWaveが使えるようになったわけだ．それとSwitcherなしでLightWaveが使えるから，メモリを4MB以上余分に使える．

LRはToasterを開発したNewtek社にとってはケムタイ商品だと思うが，LightWaveユーザにとってはまさに的を得たプロダクツになっている．

IMPRESSION

LRをToasterエミュレータとして見ると，AMIGAの数あるエミュレータの中でも“使える”部類に入る．実際に使ってみるまでは半信半疑だったけれど，LightWaveスクリーンが現れ，しばらく使っているうちに安心感が生まれてきた．Toasterボードを付加させて使っているのと全く同じ使用感だ．システムがハングアップすることもないし，すべての機能も正常に作動する．これなら仕事で使っても安心だ．

数台のAMIGAを使い，その中の1台にToasterボードを差す．他のAMIGAでは一つのLRを共有する．Toasterボードが付いたAMIGAでメインの作業をし，他のAMIGAをレンダリング専用にする．こうすることで，これまでの何倍(マシンの数に比例し)も作業効率が上がるというわけだ．

こんなプロダクツがリリースされるとは誰も予想しなかっただろう．

だからAMIGAは面白い．

●**Light Rave** (Warm & Fuzzy Logic) 68,000円

(パイナップル調べ．すべて税別．()内はメーカー名)

■今回，紹介した製品，その他AMIGAに関するお問い合わせは，編集部AMIGA係(必ず手紙またはFAXで)またはパイナップル(℡03-3294-6502，FAX03-3875-9726．月曜日を除く13：00～20：00)まで，お気軽にどうぞ．

INSIDE AMIGA(No.5)

まず，筆者の勝手な理由で，前回このコーナーを休んでしまったことをお詫びします．

●

前回は2台目のHD(ハードディスク)を搭載する場合のジャンパの切り替えまでを紹介したので，今回はHDのプレップ/フォーマット方法を紹介しよう．

A4000に2台のHDを搭載しただけでは，まだそれらのHDはまだ使えない．1台目のHDがプレップ/フォーマット済みならば(つまりA4000を買ったときに内蔵されてくるHD)ならば，そちらのHDからシステムを立ち上げられる．もし2台とも新しく付加させた場合はA4000にバンドルされてくるFDの内の1枚“INSTALL3.0”を使ってシステムを立ち上げよう．

Workbenchスクリーンが表示されたら，2台目のHDアイコンが画面上に表示されているかを確認しよう．表示されていてもされていない場合でも，とりあえずSYS:ディスク(ブートに使ったデバイス)の中にセーブされている“HD TOOL BOX”(以下，HTB)を実行させる．普通，ブートしたデバイスがHDだった場合，HTBは“TOOLS”ドロワーに，FDだった場合は“HD TOOLS”ドローワーの中にセーブされている．

HTBを実行させてスクリーンがオープンしたら，画面真ん中にあるウィンドウの中にHDの名前(メーカー名など)のインフォメーションが表示される．もし，2台のHDのどちらかのインフォメーションしか表示されていない場合は，ウィンドウの“STATUS”を見る．ほとんどの場合，STATUSが“CHANGED”になっていると思う．

●

次にスクリーンの右下にある“SAVE CHANGES DRIVE”をチェック．ここがゴースト，つまり文字が網掛け状態になっていない場合には，そこをマウスでクリック．そうすると“SAVE～”がゴーストになり，“STATUS”がNO CHANGEDに変わる．それを確認したら，いったんHTBをEXIT．そしてリセットをかけ，Workbenchをリブート．

今度はWorkbenchスクリーン上に二つのHDアイコンが表示されるはずだ．しかし，これではまだ新しく付加させたHDは使えない．そこでもう一度HTBをオープン．今度はパーティションを決める．

まずパーティションするほうのHDをマウスでセレクト．つぎに“Partition Drive”をマウスでオープン．パーティションが一つだけでOKならばここをEXITし，前述したように“STATUS”をNO CHANGEDにし，HTBをEXIT．パーティションを二つ以上使いたい場合には，“NEW Partition”を使って，いくつかに分ける．

A4000に付加させた2台のHDのメーカー，型番が同じ場合は，ファイルシステムを“FAST FILE SYSTEM”にしておけばOKだが，1台目が“Seagate”社製，2台目が“Quantum”社製の場合，“Quantum”の方を“CUSTUM FILE SYSTEM”に設定する必要がある．それ以外のファイルシステムでQuantumドライブをフォーマットすると，一見ちゃんと動作しているフリをするけれど，セーブしたデータがまともにロードできないなどのアクシデントに見舞われるだろう．

CUSTUM FILE SYSTEMのパラメータは，Identifier=0x444f5303，Mask=0xfffffffe，MaxTransfer=0xffffffに設定するのが，いまのところベストだ．筆者もこれでフォーマットして使っているが，ほとんどエラーは発生していない．それでも調子が悪い場合は，たぶんPAULAチップとHDとの相性が悪いのかもしれない．もうご存じだとは思うが，A4000に搭載されているカスタムチップの中でPAULAだけが昔のままだ．PAULAはI/O関係をコントロールするカスタムチップなので，何かしらの悪影響をおよぼしているはずだ．これはまだ推測だが，新しいPAULA(32ビット)がリリースされたら，それと差し替えればこの問題はたぶん解消されると思う．

●

さて，話を元に戻そう．パーティションを決めたらあとはデバイス名(DH2，HH0など)を設定し，フォーマットするだけだ．フォーマットは，HTBをEXITし，Workbenchまたはシェルから行うのが確実かつ安全だ．

Workbenchで行う場合は，まずフォーマットしたいディスクアイコンをマウスで選び，プルダウンメニュー“Icons”の中にある“Format disk”を使う．シェルから行う場合には，以下のコマンドをキーボードから入力する．

FORMAT DRIVE DH0: (HTBで設定したデバイス名＋:)
NAME xxx (ここには好きなディスク名を30字以内で入力)
NOICONS (トラッシュカンがいらない場合)

●

入力が終わったら，リターンキーを押して実行する．これでHDDを使えるようになる．

スペースの関係上，あまりくわしく説明できなかったが，またこのコーナーで皆さんの質問，疑問に答えていきたいと思うので，よろしく．

では，また．

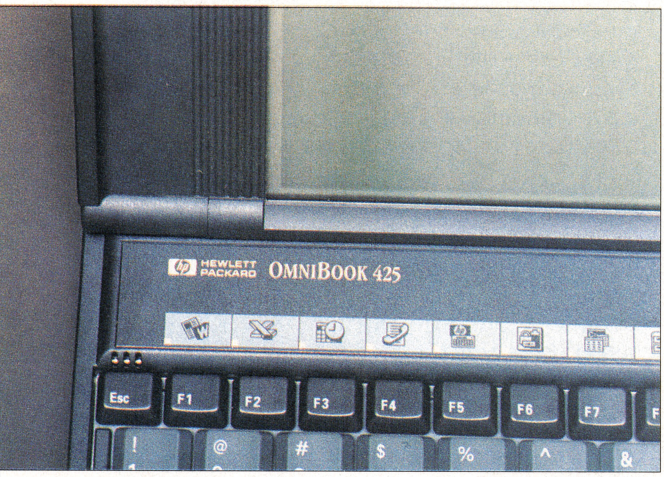
(桃園隆夫)

no.28

COMDEX Fall'93 & Solaris+WABI

今年もCOMDEX/Fallへ行ってきました!! ただし，感想は「ハズレ」. (^^;
今回はなぜハズレだったかという理由と，
帰国後，遊んでしまったSolaris 2.1 for x86+WABIのお話でもしましょう.

●著者
西川和久
Kazuhisa NISHIKAWA



これがOmniBook425

面白くなかったCOMDEX Fall'93

まず，お詫びしなければならないのが，

「今回は写真を一枚も写していません」

なぜか写したくなるような被写体が一つもなかったのです．(^^; たしかにPCMCIA，PCI，Plug and Play，PowerPC……，いろいろとモノは出ていたのですが，何かいまさらの感じがあり，これは凄い!! と感じられたモノは皆無でした．

メインホールの主な展示は，アプリケーションのセット販売=suiteと呼ばれる巨大なアプリケーションパッケージと，PowerPC，αPC，R4000/4400など，RISC PC vs Intel PCのWindows NT攻撃，これが一番印象的でした．もちろんビデオボードメーカーも片っ端から回ってみましたが，最近日本国内でもかなり速く情報が入ってくるので，特に目新しいボードを発見することはできません．マルチメディアも画質などに違いはありますが，今や動画は当たり前……．とにかくパッとしません．

さらに，ガッカリしたのがMicrosoft．“Microsoft Office”（Word，Excel，PowerPoint，Mailのセット）を大々的に展示して，お目当てのChicago，Cairoは影も

形も，それどころかWindows for Workgroups 3.11でさえも非常に小さい一コマでこっそり展示していた程度なのです(**表1**).

このような中，傑作だったのはPlug and Playのコーナーで展示していたWindows．実はChicagoだったのですが，説明を聞いても決してChicagoとかWindows 4.0などとは言わず，「WindowsのPlug and Play版」「Windowsの次バージョン」と表現しているのには笑いました．起動時のタイトルロゴまでが“Windows for Plug and Play Demo Version”．よほどNDA(Non-Disclosure Agreement)が厳しいのでしょね!! :-) 見れば，見るほどOS/2 2.1そっくりです．アイコンのデザインがMicrosoft風になり，少し趣味のよいOS/2って感じです．

ほんの数日前にVisual C++ 1.5のアナウンスがありましたが，これも展示はありませんでした(**表2参照**)．1週間程度のズレなら見せてくれてもよさそうなものなのに……．:-< しかし，気がつくと，また日本語版は一つ遅れのバージョン……．このタイムラグ，早く何とかしていただきたいものです．

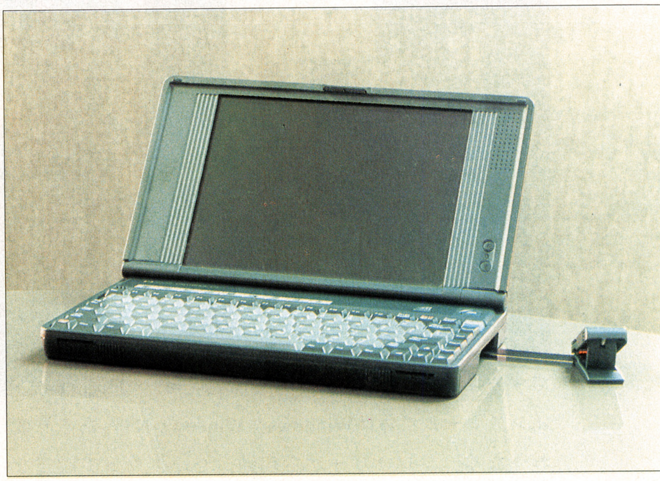
そのほか，Hewlett-Packardのブースには OmniBook425(**写真1**)が展示してありました．'93年11月号で紹介したOmniBook300と構成はそのままで，CPUがCyrix 486SLC/25MHzになっただけで，実際は少し速くなった程度のようです．ただし，これによって旧モデル(?)となるOmniBook300の値段がかなり下がりました．:-) 日本価格もあわせて下がることを期待しています．>YHP

以上，非常に簡単ですが今回のCOMDEXレポートです．面白くなかったせいか，記憶をたどってもあまりネタが出てきません．お許しください……．(^^;

表1●Windows for Workgroups 3.11の特徴

- ローカルディスクアクセスが50%パフォーマンスアップ
- ネットワークディスクアクセスが100%パフォーマンスアップ
- MS-DOS 6.2のDoubleSpaceドライブへも32ビットファイルアクセス可能
- Microsoft At Workに準拠した，FAXソフトウェアが標準装備
- システム管理が容易になった
- Remote Access Services(RAS)装備
- 32ビットIPX/SPX互換トランスポートをNetBIOSとともにサポート

写真1●OmniBook425：左ページのロゴ部分を別にすれば，外観からは300か425かの区別はできない(ただしOmniBook425の写真は，COMDEX会場でではなく，筆者事務所で撮影したもの)



Solaris 2.1 for x86 + WABI

ちょっとしたことから，Solaris 2.1 for x86 + WABI(Windows Application Binary Interface)が会社にやってきました(**写真2**，**写真3**)．UNIXには弱い筆者

表2●The new Visual C++ Product Family

●Visual C++ 1.5 Professional Edition
- DOSアプリケーションの作成が可能
- WindowsとWindowsNTの両環境で動作
- 16ビットOLE 2.0とODBCデータベースアプリケーションの作成が可能
- $599
- Visual C++ User Upgrade $99

●Visual C++ 16/32-bit Suite
- “Visual C++ 1.5 Professional Edition” + “Visual C++ 1.0 32-bit Edition”
- 16ビットと32ビットのアプリケーションを作成可能
- $749
- Microsoftのデータベースと開発ツールユーザのUpgrade $199

●Visual C++ 1.0 32-bit Edition
- Windows(Win32s使用)とWindowsNTのアプリケーションを作成可能
- $599
- Win32 PDKとVisual C++ Professional Editionのユーザは'93年12月31日までUpgrade $99，以後$139

写真2●Solarisのパッケージ内容
日本語版は発売未定





ですから，かなり不安はありましたが，インストール自体はWindowsやWindows NTとあまり変わりなく，フロッピーから起動，ハードウェア設定などを行い，CD-ROMから流し込むだけなので，比較的簡単です．

マシンの関係上，CPU i486DX/33MHz，メモリ16MB，AHA1542B，ET4000 VL-Busの構成にしましたが，SCSIアダプタ，ビデオボード，サウンドボードはMicrosoft系のOSとは違い，ハードウェアがかなり限定されますから，注意が必要です．あとで分かった話なのですが，ビデオドライバがどんなビデオチップを使っても，アクセラレータ機能を一切使わず，単にVRAMへ直接書き込むだけなので，VRAMアクセスが速いハードウェアを使った方が無難です．そういった意味から，ET4000のVL-Bus版を偶然使ったのは正解でした．

さて，マシン構成からも「かなり重いのでは!?」と勝手に思い込んでいましたが，実際使ってみると結構サクサク動きます．これには正直驚きました．いままで，32ビットOSはOS/2 2.1とWindows NT程度しか触ったことがないので，Solarisは未知の世界だったのですが，/usr/openwin/bin/openwinで，OpenWindowsを実行して触ってみると，同じ環境で使うOS/2よりも軽いぐらいです．ディスプレイも安物の14インチを使いましたが，不思議といつも見慣れている1,024×768/256色の画面が妙に奇麗なのです．GUIに品があるとでもいいましょうか…….ネットワークには接続せず，スタンドアロンなのでUNIXの醍醐味は味わえませんが，少し遊んだ感じでは，面白そうな環境です．

表3●WinTach1.0でのベンチマーク結果（1,024×768/256色）

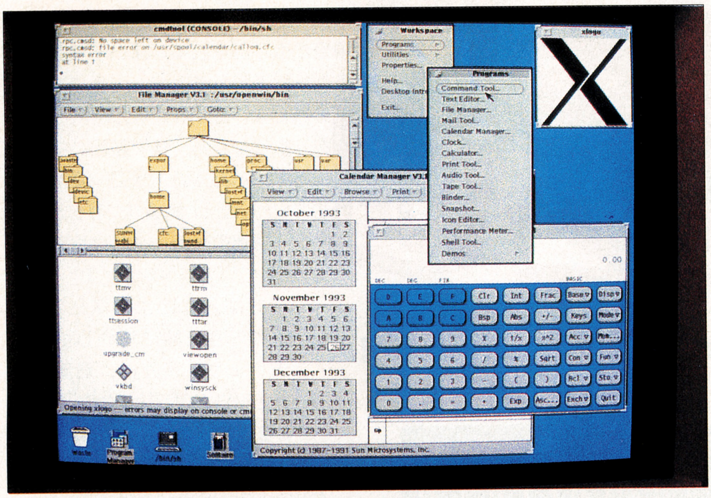
・Word Processing	1.22
・CAD/Draw	11.00
・Spreadsheet	5.01
・Paint	2.81
・Oveall	5.01

＊WinTach1.0：Texas Instruments社のWindows3.1用ベンチマークプログラム

●

さて次はWABIです．Windows APIをX11 APIに翻訳し，Windowsアプリケーションを実行するWABIは以前から筆者が注目していたWindows環境のひとつです．なぜなら，主に仕事で使うアプリケーションがWord，Excel，秀丸，秀Term程度なら，純正Windows環境にこだわる必要もなく，見た目がWindowsよりかっこよければ，どんな環境でもかまわないからです．:-)　ただし，「WABIがあるから，SolarisユーザはWindowsを買う必要がない!!」と思うのは早合点で，WABIをインストールした後に“Windows Install”プログラムを使って，必要なファイルをWABIの管理下へコピーします．これは，WABIがあくまでも“API翻訳者”であり，Windowsシステムそのものの互換環境ではないためです．

ホームディレクトリへ./wabi/windowsというディレクトリを作り，そこへWindowsのファイルをそっくりそのままコピーします（他にも/wabi/autoexec.bat，/wabi/fc，/wabi/tmp，./wabi/wabihomeなどのファイル/ディレクトリを作ります）．SoftPCのように，DOS用の



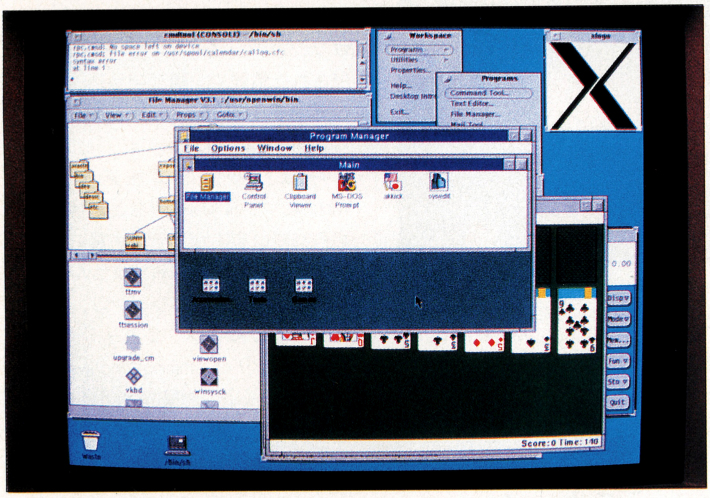


写真3●Solaris画面写真

（写真上）筆者にとって未知の世界だったSolaris．実際に動作させてみると，OS/2より軽いように感じられた．写真はもちろん，OS/2のシームレスウィンドウではありません，念のため:-)

（写真下）次にWABIをインストールした画面．いつも見慣れた1,024×768/256色名のに，なぜだか上品な感じがするのはなぜだろう？

巨大な仮想ファイルを作り，その中にFATファイルシステムをエミュレーションするのではなく，UNIXのファイルシステムのままディレクトリやファイルを作るので管理も楽です．また，フロッピードライブなども，WABI上からでは普通にA:やC:とDOS風のアクセスがそのままできますので，相互間のファイルコピーも非常に簡単で，UNIXを意識する必要はまったくありません．

さて，肝心の実行速度ですが，テストした環境では「ちょっと辛い」というのが実感です．参考のためにWinTach 1.0のデータは**表3**の通りです．

ちょうど，Windows 3.0が出た頃の懐かしい値が出ます．(^^; 最近の環境と比べると，1/10程度のパフォーマンスでしょうか!? たしか，今年のデータショウで日本IBMが展示していたPowerPCでは，WABIを使ってもi486DX2/66MHz+Mach32並みの値を出していましたから，今回のテスト環境がかなり悪いのかもしれません……．いずれにしても，もう少しパワーのあるマシーンが要求されるのには違いがなさそうです．

●

次回は間に合えば，もうひとつのPC UNIX "NEXTSTEP 3.2"（SoftPC+Windows）環境をご紹介します．:-)

〈DOS/Vに関する疑問・質問にお答えします〉

■本コーナー，およびDOS/V全般に関するご意見・ご質問を筆者が直接お受けします．ご質問のある方は，「DOS/Vパラダイス質問係」と明記の上，巻末の愛読者カード，またはBBSの以下のIDまでメールでお寄せください．本コーナーの中でご回答致します（直接メールなどで回答を差し上げることはありません）．

●NIFTY-Serve: PFG00112●日経MIX: knishika●PC-VAN: DDE34670

CD-ROM天国 第3回

●今月のお題●

Microsoft Developer Network CD

(Fall 1993 / Microsoft)



●このCD-ROMの簡単な紹介●

今回は，プログラムを組む人にはぜひ持っていてほしいCD-ROM，Microsoft Developer Network CD(MSDNCD)を紹介しましょう．このCD-ROMは，Microsoftのアプリケーションや言語，それに関連する標準規格の情報などが山ほど詰まったCD-ROMです．

●著者●

えるちゃん

ここしばらく，いくつかのパソコン雑誌でこのCD-ROMについて紹介されていたことがあるので，すでにご存じの方もおいでだと思います．これまでは日本からはこのCD-ROMを購読することはできませんでしたが，やっとのことで，日本でもこのサービスを受けられるようになりました．すでに申し込みは開始されています．年間2万9千円の費用を払い込むことで，年に4枚の毎回アップデートされた情報の入ったCD-ROMが届きます．CD-ROMのほかには，隔月で英語版のニュースレターが届きます．加えて日本では，英語版にひと月遅れて日本語に翻訳されたものが届くことになっています．

MSDNCDは，マルチメディアビューア用のファイルとなっています．マルチメディアビューアは，Microsoftの他のマルチメディアアプリケーションでも採用されているビューアです．MSDNCDのファイルサイズは300MBを超える巨大なものとなっています．

MSDNCDのセットアッププログラムを実行すると，Windowsのあるディレクトリにビューアのほかに数本のファイルがインストールされます．セットアップには3種類あり，フルセットアップではハードディスクの領域を約19MB使用しますが，その分情報の検索は早くなります．最小のセットアップでは，100KB程度のディスクしか使用しませんが，情報はすべてCD-ROM上から読みだすので，検索速度はがた落ちになります．

●使いやすい二つのウィンドウ

MSDNCDは二つのウィンドウで構成されています．実際の内容が表示されるメインウィンドウと，記事のインデックスを表示するContentsウィンドウです．互いのウィンドウには，相互に行き来するためのボタンが用意されているので，いちいちウィンドウをクリックすることなく，ウィンドウの切り替えができます．

それぞれの項目にサンプルがついている場合には，メインウィンドウの文中にクリック可能なアイコンが表示されます．複数のアイコンが文中に表示されることもあります．アイコンをクリックすると，そのサンプルに含まれるファイルの一覧が表示されます．その一覧の中からコピーしたいファイルを選択し，コピー先を指定することで，独立したファイルとして別のディスクに保存することができます．

検索機能は条件付けをした検索も行えるようになっています．複数のキーワードをAND/OR/NOTを組み合わせた形で検索を行えるほか，あるキーワードからユーザが指定した範囲にあるキーワードを検索することもできます．収録されている情報が膨大なため，単純なキーワードでは，1,000個単位の項目が取り出されてしまいます．複合条件検索を行うことで，上手に絞り込みを行わないと，なかなか目的の情報が取り出せません．検索範囲の指定も可能です．Windows3.1のSDK/DDKの情報の中からだけ検索を行ったり，VisualC++関連の情報からだけ検索を行えます．

では，なにより気になる，その内容を順に紹介していきましょう．今回は，最新版であるMSDNCD-Vol5の内容を紹介していきます．

●10のセクションから構成

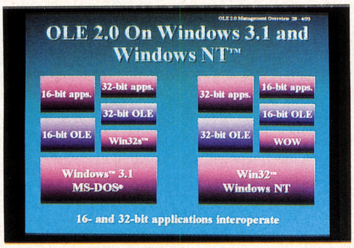
MSDNCD5は大きく10個のセクションに分かれています．まず，**WelCome**セクション．ここでは，そのCDで追加された情報や，購読申し込みのためのフォームが用意されています．

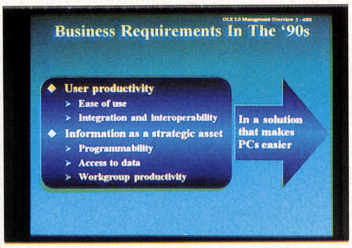
TechnicalArticlesには，TechNoteとと呼ばれる技術情報を中心に，WindowsやLAN Managerなどの情報が収められていますが，そう大量の情報が入っているわけではありません．

SampleCodeセクションは，製品に付属してくるサンプルコードのほか，Petzoldの“ProgrammingWindows”のサンプルコード，“MicrosoftSystem Journal”の記事に掲載されたソース，Microsoftが公式にリリースしているプログラム開発のためのソースまでもが収められています．

Specs and Strategyのセクションは非常に有用です．このセクションは二つのサブセクションに分かれています．Backgroundersセクションでは，製品や規格に関する概略が説明されています．代表的なものでは，AtWorkやMFC，OLEなどが解説の対象になっています．このほかにもソフトの検査を自動的に行うツールであるMS-Testの説明や データベース接続の規格であるODBC(Open Database Connectivity)の説明も入っています．もう一つのサブセクション，Specificationsセクションでは，いろいろな規格をはじめとする情報の詳細な情報が入っています．DPMI(DosProtectedModeInterface)のスペック0.9やNDIS(Network Driver Interface)

●著者近況：CD-ROM道を極めんと研鑽を積む毎日．しかし，CD-ROMのほかにもあんなことやこんなことにも夢中で，なんだかとっても大変らしい．





◀写真はいずれもSpec and Strategy/Backgroundersのセクションから，Windows 3.1/NTのOLE 2.0に関する解説の画面

のスペック2.0.1をはじめとして，このほかにも多くの規格に関する説明が入っています．面白いのは，OS/2からWindowsにプログラムを移植するためについての詳細な文書がついていることです．このようなノウハウが網羅的にまとめられている文書はあまりみかけませんから，結構参考になります．WindowsからOS/2にプログラムを移植する場合にも逆に参照することもできるでしょう．

●Unicodeの規格書

MSDNCD5になって，Unicodeの資料が付属するようになりました．Unicode Inc.が作成した規格書そのものであり，そのバージョンは1.1です．規格の作成元から公式に出された情報はその規格に沿って物を作る場合には押さえておきたいものですから，大変ありがたいものです．

今後発売されるDOS6に関する情報としては，ディスク容量を倍増するためのツール，DblSpace関連の情報が入っています．圧縮ファイルのファイルフォーマット・圧縮ファイルを扱うためのAPI・そしてファイルのリアルタイム圧縮インターフェースの資料までが収められています．

Knowledge Base and Bug Listsが有用です．製品のバグに関する情報が中心ですが，そのほかにも関連情報などもまとめて収められています．

Books and Periodicalsには，開発関連書籍が数多く入っています．入っている書籍名を挙げていくと，Advanced MS-DOS Programming，Visual Basic Programmer's Journal Articles，Cobb Group Visual Basic Articles……．これだけの書籍を店頭で購入する費用を考えれば，それだけで年間購読する価値があります．さらに，一部ではありますが，これらの書籍には電子媒体ならではの処理が加えられています．文章中に出てくるWindowsやC，BASICの関数名などをクリックすると，その解説を表示することができます．

ProductDocumentationセクションは製品のマニュアルが入っています．サブセクションとして，言語，アプリケーション，SDK，DDKの四つのセクションが用意されています．結構貴重なのはDDKのセクションでしょう．DDK自体高価ですし，あまり見る機会のないものです．しかし，デバイスドライバを記述するためには，よりWindowsの内部に近い部分に触れなければなりませんが，そのための情報をこのような形で入手することができるのは，実際に開発を行っている人にはとてもありがたいものです．言語マニュアルには早くもVisual C++の32bitバージョンのマニュアルも入っています．アプリのセクションには，ExcelのSDKやMS-WordのマクロのSDKマニュアルまで入っています．

ツール群は二つのセクションに分かれています．公式にサポートを行うツールと行わないツールです．公式にサポートを行うツールの中には，市販もされているResourceKitが入っています．日本ではあまり使われていないと思われるResourceKitですが，Windows，Windows for Workgroup，Windows NTそれぞれのResourceKitが入っているだけでもうれしいものです．

特にサポートを行わないツール群は結構役に立つものが入っています．プログラムの自習用モジュールとが五つ入っています．すべてWindows用ですが，コモンダイアログ，MDI(Multiple Document Interface)，DIB(Device Independent Bitmap)，パレットの操作，印刷を自習することができます．

最後のConference and Seminarsセクションは見た通り，米国内で行われたセミナなどの資料が収められています．実際にプレゼンテーションに使用した(と思われる)PowerPointで作成されたデータと，専用ビューアがセットになって入っているので，実際のプレゼンテーション画面を自分の手元で見ることができます．セミナの内容のテキストの中にはWindows for Pen用のユーザーインターフェースの設計方法や，MS-Wordのマクロ言語でもあるWordBasicに関する情報なども入っています．日本ではあまり情報が出まわっていないことについてもいろいろなことが記述されています．

●

とりとめもなくMSDNCDの，そのほんの一部を紹介してきました．収められている情報量があまりにも莫大で，筆者もまだその全部のドキュメントに目を通せていません(まぁ，全部に目を通す必要もないのですが)．MSDNCDは書籍として読むもよし，オンラインリファレンスとして使うもよしの非常に便利なCDです．DOS/Windowsでの開発を行っている人はぜひこのCDを持っている人を探してでも一度目を通してみてください．きっと自分でもほしくなることでしょう．

〈MSDNCD：補足〉

MSDNCDは，Microsoft Developer Networkの会員に対して届けられるものです．入会方法など，くわしくは**マイクロソフト デベロッパー ネットワーク事務局**にお問い合わせください．TEL 03-5461-2617(10:00～17:00．祝祭日を除く月～金)／FAX 03-5461-2627

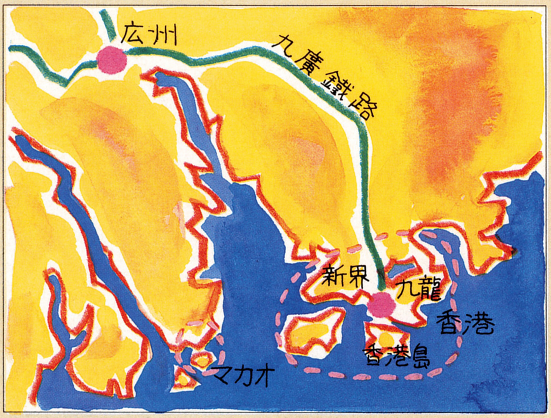
広州・香港電脳案内

～Macintoshを中心として

❶ 廣州電脳城編

昨年の5月，北京のコンピュータ状況を取材した(本誌'92年9月号・北京電脳案内2「泥融飛燕子」を参照)．今回，その後の中国大陸のコンピュータ事情をレポートしたく，広州に向かったのだが，広州の電脳中心「廣州電脳城」を取材しているうちに，中文文化圏のMacintoshの状況がどうなっているのか，興味がわいてきた．今回と次回の2回に渡り，Macintoshを中心に広州・香港の電脳事情をレポートする．

●著者
東　均
Hitoshi HIGASHI



広州・香港概略図

9月×日，香港・九龍駅から九廣鐵路(KCR)の列車でゆられること3時間弱，経済特別区・深圳を越えて広州市に着いた．

広州市は，中国の南大門，珠江の河口近くにある広東の州都，南方最大の都市である．別名，羊城という．周の夷王時代，飢饉に悩む人民のために，5人の仙人が苗をくわえた5匹の羊にのって訪れたことに由来する．のち仙人は空に消え，5匹の羊は五つの石となった．記念の五羊石が越秀公園にある．

晩夏の広州は，香港よりも暑く，盲流として有名になった人の海ではなく，車の連珠，大洪水であった．5kmほど離れた目的地，廣州電脳城に到着するまで一時間近くかかった．

・・・・・廣州電脳城・・・・・

広州には，北京の中関村街のようなコンピュータ街はなく，20くらいのパソコン店舗が入ったビル「廣州電脳城」がある．タクシーから降り，さっそく廣州電脳城に向かい，地下の店舗に入る．

●金百合電脳通訊有限公司：登祺燊

登祺燊「広州には，中華人民共和国製のパソコンはないでしょう．長城(Great Wall)や聯想といったコンピュータは評判が悪くて置いていません．政府は国産品を使うことを勧めていますが，性能が悪く，値段も安くはないのです．廣州電脳城にあるパソコンは，すべてアメリカ・台湾経由のパソコンです．ここに置いているコンピュータ？　ASTリサーチの製品です．売れ筋は486/33MHz，HD240MBで25,000元(約425,000円．1元は約17円)です．好きなパーツをアセンブルできます．Windows3.1は，英語版が動作しています．中文？　このコンピュータは中文のシステムを組み込んでいないので，表示はできません」

ASTリサーチのコンピュータを動かしてもらう．

一片石

Super-CCDOS5.00と表示され，DOSが起動した．DOS上で一番使われている中文ワープロ“Super-Wps文字処理系統”を見せてもらう．Windows3.1は，英文だけのときによく使われ，中文用にはあまり使われないとのことである．

去年，取材した北京のコンピュータの主流は，386でCC-DOS(Chinese Character DOS)3.3であった．今年は，486に移行し，DOSも5.0になったようである．北京で見ることができなかった中文ワープロ“Super-Wps文字処理系統”を見ることができたのはうれしかった．しかし，国産のコンピュータは，広州では使われていない．

登祺榮「ASTリサーチ社は，天津でパソコンを製造することになると思います．中国全土で販売し，中国本社施設も建設する予定です．ASTリサーチは，中国のPC市場の1/4を支配しています．Compaqも深圳でデスクトップコンピュータの製造工場を建設する予定だと聞いています」

登祺榮氏の取材を終えて店を出ると，DELLという文字が見えた．DELLは，アメリカのコンピュータメーカーである．さっそくドアを開く．

●DELL 中国指定代理商 摩力克電脳公司：楊暉

楊暉「ここはDELLの代理店です．廣州電脳城にはDELLのほかに，Compaq，ASTリサーチなどのコンピュータが販売されていると思います．広州には中華人民共和国のパソコンはないでしょう．デモしているコンピュータは，DELL 4033/XEです．8MバイトのRAM，HDが2GB(SCSI)です．値段は42,800元(約727,600円)．中文フォント？ 簡体漢字，繁体漢字合わせて50種類以上あります．簡体漢字では，本宋，宋三，大標宋，黒体，大黒，粗円，水柱，姚体，行楷などがあり，繁体漢字では，秀麗，書宋，中楷，平黒，黒体，細円，彩雲，行楷などがあります．すべて北京大学新技術公司作成のフォントで，TrueTypeです．PostScriptフォント？ 知りません．

DELLのコンピュータを購入するのは，ほとんど公司(会社)です．まれに学生が購入する場合もありますが，386を親に買ってもらうのがほとんどです」

個人ユーザは，去年北京でレポートしたときと変わらず少ない．華南経済圏で高所得者が増えているといわれているが，状況はあまり変わっていないようである．

しかし，華南経済圏を含む沿海地方の発展によって中国のパソコン市場は年率25%の割合で成長している．ASTリサーチやCompaqは，中小企業や一般人にも普及する巨大な市場となることを見越し，中国本土への進出を行っているのだ．とくにASTリサーチは，中国市場で実績がある先駆的存在である．中国市場は活気づくだろう．

なお，広州では，香港ドルが流通していた．北京で外国人が使わなければならない兌換券は，見たこともなく，受け取ってくれない．人民元と香港ドルとの比率



金百合電脳通訊有限公司の登祺榮さん

亜洲電脳事情シリーズ

広州・香港電脳案内

～Macintoshを中心として

❶ 廣州電脳城編

は，0.9:1であり（闇のレートでは1.1:1.0と逆転しているようだが），香港ドルでの支払いを喜ぶ．中華人民共和国は，華南経済圏に浸食されているようだ．本国製のコンピュータが皆無であることだけでなく，香港ドルの流通によって，その一斑一端が分かる．

それにしても勇躍雀躍して，広州入りしたが，中華人民共和国製のコンピュータは見当たらない．アメリカ製や台湾製のパソコンを使っているのであれば，レポートする必要性はあまりない．さまざまなパソコン雑誌で，香港や台湾のIBMコンパチマシンはレポート済みである（もっとも文化的側面のレポートは少ないが）．中華人民共和国製コンピュータのその後をレポートしようとした目論見は潰えた．廣州電脳城を見て回ると，1店舗のみMacintoshの販売店がある．しかし，マシンの展示はない．中に入って話を伺うと，広州でMacintoshを使って仕事をしている公司（会社）があるという．

昨年，北京の中関村街を訪問したときに，Macintoshを販売している店舗はまったくなく，Macintoshで仕事をする会社など皆無であったろう．予定を変更し，Macintoshの中華人民共和国内での状況を知りたくなっ

廣州電脳城入口



羊公一片石

た．まずは，Macintoshの販売店で伺った威仕系統(WYSSYSTEM)という会社へ向かう．黄花崗公園のとなりである．黄花崗公園には，1895年，孫文が指導した広州起義の犠牲者たち(その多くは日本留学経験者)が眠る七十二烈士墓がある．

●威仕系統(WYSSYSTEM)：任暁峰

黄花崗公園の近くの壁に，Macintoshの蘋果(りんご)マークがあった．通りを入るとすぐ威仕系統の文字が見え，すぐ分かった．マネージャの任暁峰氏にお話を聞くことができた．

任暁峰「威仕系統(WYSSYSTEM)は，中国大陸と香港の合弁会社です．主な業務は，印刷会社や広告会社の商品のデザインです．Macintoshで卓上排版系統(DTP)を行っているわけです．総勢20人ぐらいのスタッフでやっています．1冊の本をそのまま作成することは少なく，広告用のカラーポスターや製品カタログをDTPで行っています．TurboPS/660Bというレーザープリンタでカンプ*を作成し，フィルム出力をLinotronic 200SQで行っています．もうすぐ，最新のLinotronic 330RIP50が広州でも導入されます」

任暁峰氏は，ハンサムで背が高く，物腰がやわらかい．インテリの風情があり，中国のエリートの一人であろう．さっそくMacintoshを見せてもらう．Quadra800以下，LCⅢなど10台近くのマシンが稼働していた．Quadra800では，Illustrator3.2が動作し，グラフィックデザインを行っていた．

任暁峰「中文の繁体字，簡体字の表示と出力？ できます．インストールしているシステムは，ChineseSystem 7.1Tです．Tは，Traditonal Chinese(繁体字)の略です．ChineseSystem7.1Tは Multi-Language Systemです．GBコードのBeijing(北京)Fontを輸入方法(Input Method)で表示し，Big5コードのTaipei(台北)Fontを，大易・注音などの文字入力方法で表示します．双方に簡体字と繁体字があり，できればコードを統一してほしい．2種類のコードが存在するのは，中華人民共和国と台湾とのPolitical Problemです．しかたなく，自分でコード変換プログラムを作成しました．ご覧になりますか?」

*カンプは，完成品を作る前の，デザインの完成予想図のようなもの

威仕系統(WYSSYSTEM)入口の看板



Codestar2.04という名のプログラムは，テキスト形式のファイルで変換を行う．GBコードとは，中華人民共和国が制定したコード体系(国家標準［Guojia Biaozhun］)であり，Big5コードとは，台湾が設定したコード体系である．つまり，同じ中文でコードが異なるため，GBコードのフォントをBig5コードのフォントに変更すると文字バケが起こる．文字バケを避けるためには，いったんGBコードの文字を保存し，Codestar2.04を通してBig5のコードの文字に変換し，それを読み込むという手順を踏むことになる．

Codestar2.04はC言語で作成したという．販売すると儲かるのではないかと問うと，必要な人は必要でしょうが，まああまり売れないかもしれないので販売する予定はありません，と言う．あまり欲がなさそうな人である．“銭向”(お金もうけ)が一般的になっている中華人民共和国でまれな人のように思われる．

任暁峰「使用しているソフトウェアは，PageMaker3.5CT，CTはChinese Traditionalの略です．そしてIllustrator3.2です．グラフィックデザインは，ほとんどIllustrator3.2で行います．しかし，Illustrator 3.2は，中文が使用できません．中文の処理には，FreeHand3.1Aを使い，EPSファイルでやりとりしています．QuarkXPress? QuarkXPressは，中文の使用が難しく，使っていません．あと，Photoshop2.5を使います．Photoshop2.5も中文の処理はできません．PageMaker3.5CT，Illustrator3.2，FreeHand3.1A，Photoshop2.5で仕事をまかなっている感じです．

亜洲電脳事情シリーズ

広州・香港電脳案内

～Macintoshを中心として

❶ 廣州電脳城編

日本ではPageMaker4.5Jが動作している？ そうですか，早くPageMaker 5.0CTを出荷してほしいです．現時点のDTPの問題点は，中文文字の字詰と分色です．画面の表示と出力が異なり，色が思ったように出ません．経験を積むことで回避している状態です．フォントの数？ まあ，現時点で最低限がまんできる程度にはあります．多ければ多いほどよいですが……．

具体的な中文の入力ですか？ GBコードを入力するときは，輸入方法（Input Method）で行いますが，これには五筆型，五筆画，拼音，区位，羅馬の5種類があります．Big5のときは，大易，注音，拼音，倉頡，半形鍵盤，漢音の6種類で行います．一番速く入力できるのは五筆型です．え？ 拼音？ いえ五筆型です」

中華人民共和国では，1957年，漢語拼音（ピンイン）方案が策定され，ローマ字を利用する発音表示が提唱された．つまり，ローマ字表記によって漢字を表示させる方式が，一番速い中文入力方法だと思ったのである．

任暁峰「拼音入力は，中華人民共和国では，たしかに誰でも簡単に入力できる方法ですが，単漢字変換なのです．五筆型は，特殊な技術が必要です．私はできませんが，毎分150文字程度の入力ができます．できる人を呼びましょう」

妙齢の美女である．北京の取材時の王昭君ならぬ趙飛燕といった風情か．中国美人は目元がすずしい．さっそく入力方法を教えていただいた．“中华人民共和国”と表示させるのにKWWLとわずか4ストロークのタイプですむのには驚いた．つまり，中は口，华はイ，人民は人，共和国は，口で代用し，それぞれKWWLに対応する．この入力方法を覚えるための本が出ているという．そういえば，廣州電脳城の本屋で，『常用漢字輸入法』というビデオが発売されていたのを思い出す．中華人民共和国では，中文をすばやくタイプするための技術が重宝されるのであろう．

しかし，高速入力法がいつもベストか，という点には疑問がある．1分間150文字入力は，日本語ワープロが単に清書道具と化しているのと同じく，中文入力道具にすぎないのではないか．つまり，考えながら文章を作成し，作成しながら考えるという思考方法をとるときに，1分間150文字の入力は必要がない．“摘句老彫蟲”とまではいわないが，すばやい文字入力のみに目を奪われてもしかたがない．

句を摘むには，音とリズムが必要である．発音とリズムに直結した文字入力が自然である．中国大陸の人にとって，発音記号といえる拼音入力が最適だし，香港人にとっては，広東語の音をローマ字化した入力が最適だし，台湾人にとっては，発音記号である注音入力が最適だと思われるが，いかがであろうか．中文という一つの共通物（繁体字，簡体字の区別はあるが）を導き出すのに，発音や文化の違いによって方法がさまざまある．ここでも，“中



威仕系統（WYSSYSTEM）の任暁峰さん

羊公一片石

国，一つではない”ということを発見する．

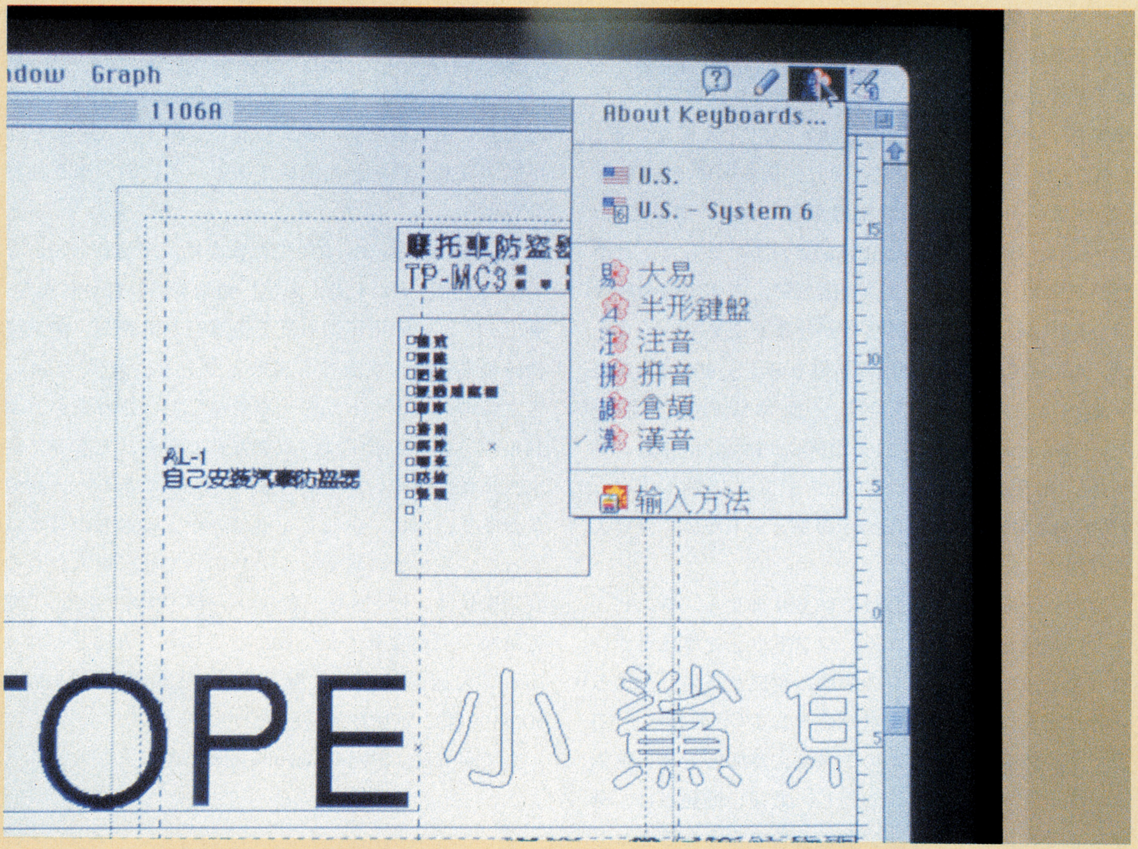
しかし，拼音入力に関して，単漢字変換では話にならない．少なくとも，連文節変換ができなければ，考えながらの文字入力はおぼつかない．PC-9800上の中国語ワープロソフト「KoaTechnoMate2中国語」（高電社）では，拼音方式と四声（音の高低を表す四つの声調記号）を組み合わせ，連文節変換を行うことができる．高電社さん，ChineseSystem7.1上の拼音連文節変換輸入方法を開発していただけませんか．できれば，KoaTechonoMate2中国語とのテキスト互換プログラム付きでお願いします．

最後に，任暁峰氏に個人的な質問をしてみる．

任暁峰「コンピュータの勉強？　大学時代からです．コンピュータ言語はCでした．その勉強が，Codestar2.04（Big5←→GBコード変換プログラム）を作るのに役立ちました．はじめに使ったパソコンは，IBMマシンです．Macintoshを使い始めたのは3年半くらい前からです．最近北京から帰ってきたばかりですが，北京にもMacintoshはあります．香港には11月30日～12月2日にMacintosh WORLD Expo/Asia'93が開催されますので，行く予定があります．ひじょうに忙しいです．おじゃまして悪い？　いえいえかまいません．そうそう私の書いた文章が『卓上時代（DESKTOP TIMES）』に掲載されています．ご覧になりますか？」

見せてもらった『卓上時代（DESKTOP TIMES）』［1993年秋季］は，任暁峰氏が所属する威仕系統（WYSSYSTEM）が香港で年4回発行するDTP雑誌であった．全ページDTPで作成し，使用しているソフトウェアは，PageMaker3.5CT，Fontographer3.5，Adobe Illustrator3.2，Aldus FreeHand3.1，Adobe Photosop2.5，Painter2.0，EPS ChineseCharacter Library2.0，Alias Sketch1.5，ColorDRAW3.0（PC），Infini-D2.0，

ChineseSystem7.1T［中文系統7.1T(繁体版)］の中文入力



PrePrint1.5，Scitex Dolev，PrePrint1.5である．イメージ入力には，Nikon NK3500，Capture4.0，HP ScanJet Ⅱc，Scitex SmartPS，フィルム出力には，Agfa SelectSet5000，Linotronic330，Rip50を使っている．

目次を拾うと，“無網分色技術”“中文TrueTypeFont”“中文System7.1”“中文排版系統（下）”“替PageMaker3.5CT做分色”“使用WACOM電子描筆”“市場動向”などがある．日本でもぜひ出版してほしいDTP関連に特化した雑誌である．

そのなかに，任暁峰氏の“Macintosh在廣州”と題したレポートがあった．要旨を紹介しよう．「3年半ほど前に筆者が広州の会社を見学したとき，ちょっと奇妙なコンピュータを見ました．小さく，モニタと一体型でそばに小さな“小老鼠”（マウス）がありました．聞くとApple Computer（蘋果電脳公司）のMacintoshというコンピュータでした．ただし，中文は使えず，英語のみの使用でした．操作は簡単で魅力的なマシンでしたが，次の問題点があると思いました．宣伝が不十分，価格が高い（PCの2〜3倍），中文が使えない，ソフトウェアが不足．しかし，問題点が解消されたらDTPの市場発展ができるだろうと思いました．広東省には，1,800もの広告関係の企業があり，広告関係で9.5億元（約161億5千万円）の巨大市場があるからです．当時，広州には，3社のMacintosh特約店がありました．華南計算机公司，広州索因軟件有限公司，広州威仕系統有限公司です．それぞれの会社が仕事を進めていくなかで，Apple Computerは国際市場を考えSystem7.1を発表しました．そして今年，簡体版の中文系統7.1（Chinese System7.1）を出荷し，Macintoshのユーザは，増えています．中文フォントに関しても，TrueTypeのほか，PostScriptのフォントが広く受け入れられました．

しかし，問題がないわけではありません．コンピュータに精通した人が少なく，ソフトウェア作成者がいないこと，ソフトウェアの説明に中文が不足していること，第三にソフトウェアが盗まれる（不正コピー）ことです．筆者は，はじめの二点は，時間とともに解消されていくと信じます．しかし，最後の問題点は，分かりません．ソフトウェア会社を殺すようなことはしないでほしい!」

取材のあと，任暁峰氏からメッセージがあった．「広州・香港にいる日本企業は，広告宣伝にぜひ威仕系統（WYSSYSTEM）をご利用ください」

Macintoshがおもしろそうである．中文文化圏におけるMacintoshの状況を知るために，急遽香港に戻ることを決意する．

●

広州では，大三元酒家で片皮乳猪（子豚の丸焼き）と八宝冬瓜盂（トウガンのスープ）を“大腹”しようと思っていたけれど，香港にとんぼ返りだ．しかし，帰りの足がない．飛行機もフェリーも汽車も，香港行きの切符がない．同行者は，大丈夫，広州駅に行けば倒売（ダフ屋）があるという．旅なれた同行者は交渉に入り，はじめ一人，HK$350（約4,550円．HK$1は約13円）から値切りに値切りHK$275（約3,575円）にする．ちなみに，正規の値段はHK$168（約2,184円）である．

香港に帰る車中，大雨に見舞われる．香港ではよくあることだそうだ．瑞雨であることを望む．夜7時，香港着．雨上がりの街を歩き，同行者と二人，日本人のあまり来ない料理屋で客家料理を賞味する．客家とは中原から南下移住した漢族である．独自の言葉と独特の習俗をもつ．孫文，登小平，リークアンユー（李光耀）は客家人である．女性では宋慶齢，宋美齢姉妹が有名．人民解放軍の元老，朱徳，葉剣英，賀竜が客家人．歴史を下れば太平天国の乱を領導した洪秀全，南宋滅亡時の愛国者・文天祥も客家人である．現代中国は，客家系の血のネットワークを無視しては語れない．葉剣英の子息・葉選平は，広東省長である．ちなみに客家人は女性が強い．その客家人の料理である．一品の量が多く，味は，少ししょっぱいが，日本人の舌には合っているように思えた．それにしても広東人たちの食事風景は，しゃべり，食らい，笑いの渦である．“健舌健啖　広東食宴”といったところか．負けじと食らい，“大腹，大腹”．値段は，盐焗鶏など4品でHK$199（約2,587円）という安さであった．

さて明日は，香港のMacintosh販売店を見て回ろう．

（つづく）

※羊公一片石（李白の詩より）　君不見羊公一片石
亀頭剥落生莓苔

1994年の展望

キーマンに聞く

新年特別インタビュー

取材・構成
本誌編集長　冨田裕一

写真・広路和夫/舟喜力



NEC
取締役支配人
高山 由 氏

「アップワードの時代」



横河・ヒューレット・パッカード株式会社
コンピュータプロダクト営業本部
パーソナルインフォメーションプロダクツ
ジャパンマーケティングセンタセンタ長
八重樫行男 氏

「モービルの幕開け」



富士通株式会社
パーソナルビジネス本部
副本部長
神田泰典 氏

「FM/Vでシェア30%」



アップワードの時代

キーマンに聞く－1994年の展望－

NEC
取締役支配人
高山 由 氏

93年度上半期の出荷台数はどのくらいですか

高山氏（以下省略）：PC-9800シリーズは58万台です．そのうちMATEとFELLOWが25万6千台で44.1％を占め，ノート型はカラーとモノクロを含めて30％ちょっとでしょうか．98MULTiは11％です．

この1年を振り返りますと，外資系のメーカーが低価格で参入してきた割には立ち上がらなかったですね．1月にFELLOWとMATEを発表し，かえって私どもに勢いをつけさせていただいたというのが実感です．といって潤ったかというと，そうじゃないです．やはり経済不況とか需要の停滞とかで，正直言って我々の期待以上には伸びていません．おかげさまでPC-9800の出荷台数は前年度並みの確保ができましたが，金額ベースでは，単価が下がっている分，ちょっと落ち込んでいます．

これはどういうことかというと，まだまだ普及率が低いということです．各家庭の普及率は，12～3％ぐらいでしょう．この数年間は横這い状態です．企業でも一人1台はほど遠いですし．低価格化だけでは，普及拡大につながらないということです．業界の工夫が必要です．

Windows3.1が5月に出荷され，半年が経過しました．出荷実績はどのくらいですか

5月に他社に先駆けてWindows3.1を出荷し，Windows時代もPC-9800だということを皆様方に理解していただきました．初期の立ち上がりは良かったですが，ここにきてちょっと横這いです．マイクロソフトで出されている数は分かりませんが，NECが出している日本語Windows3.1は累積で20万本強です．少ないですね．

思ったより立ち上がらなかった理由は，まだ魅力あるWindows3.1のアプリケーションがお客様に伝わっていないことですね．逆にいえばDOSのアプリケーションが強いということです．DOS環境ではハードディスクがなくても使えるし，メモリが少なくても使えます．そして低価格ですからね．意外でしたのは，DOS用CADがまだ売れているということです．改めて9800シリーズの強さは，これまでの豊富なソフト資産にあると認識しています．当初，FELLOW

とMATEの出荷比率は4対6でMATEが多いと予想していたのが，実際は2対1でFELLOWでした．

そこで今回発表した新MATEはWindowsの普及に狙いを定めています．新MATEには，普及機（9821Be，Bs，Bp）と高級機（9821As2，Ap2）があり，どちらもHDDモデルにはウィンドウアクセラレータを標準搭載しグラフィック機能を強化しています．特にMATEの最上位機種9821Ap2/C9TはMATOROX社のMGA-IIチップを標準搭載し，世界最高レベルになりました．従来の9821Apの2.4倍も高速です。普及型のMATEにはCIRRUS LOGIC社のCL-GD5428を標準搭載しています．今回の新製品は，根強いDOSユーザとさらに新規ユーザ獲得のための17万8千円という低価格FELLOW，企業ユーザも含めて買いやすく値段設定した26万8千円の普及型MATE．マルチメディアで高機能化されたWindows対応の高機能型MATE．これが今回の製品の３本柱です．

Windowsを普及させる戦略はあるのですか

Windowsへの移行は，DOSと共存しながらゆるやかにいくでしょう．移行へのルートは二つあります．一つはDOSを卒業した人がWindowsへ進むルート．

もう一つは，現在ダウンサイジングを図っている企業がネットワークのクライアントとして来年からWindowsを採用していくとみています．これはまちがいありません．サーバ/クライアントのネットワークになりますと，サーバにはWindowsNTやUNIX，NetWareなどのサーバ機能を発揮するものがありますが，一方端末でエンドユーザコンピューティングとなりますと，やはりWindowsが強い．そうなると，今ダウンサイジングを検討している企業の情報システムから，かなりの数で注文が入るようになります．ですからWindowsの普及は企業ユーザから急速に起こってきますよ．それを肌で感じているからこそWindowsインストール済版の新MATEを26万8千円で出したわけです．オフィス用として割り切ったので，あえて音声をオプションにしました．

企業がコンピュータへの投資をおさえている状況で，ネットワークの導入は難しくないですか

企業が情報投資をおさえているのは確かです．しかし，良く調べますとおさえているのは新規の投資であって，現在すでに使用している情報システムには金を使っているのです．月々の運用費やレンタル費を払っている．ポイントは今使用しているシステムを効率化しなくてはいけないということです．いままでシステムの運用に10の費用を使っていたとする．それをクライアント/サーバでやれば7の費用ですむとなりますと，3割の経費削減ができます．これがダウンサイジングなんですよ．従来の機能を発揮するために，ワークステーションとパソコンを使ったクライアント/サーバを導入すると経費削減ができるのです．これなら買ってくれますよ．

ネットワークが導入されると，一人1台の環境になり台数が増えますが，米国ではその消費電力が問題になっています．このときたとえばカラーTFT液晶ノート型にするとデスクトップ型の十分の一くらいの消費電力ですみます．それで最近カラーTFT液晶ノートパソコンが企業に売れているのです．まだ値段は高いですが，将来は間違いなく主流になっていきますよ．今回発表したノート型のPC-9821Ne340/Wは，340Mバイトの大容量ハードディスクを搭載して，デスクトップ型をターゲットにしたマシンです．

米国向けにはULTRA LIGHT VERSAというカラーノートを出しています．米国でのポータブルタイプは，ほとんどがセカンドマシンとして使われています．それでデスクトップ型とノート型の比は85対15ぐらいです．日本ではこの比は65対35になり，しかもノートタ

イプは最初のマシンとして買うという状況です．

そこで私は、米国でデスクトップ型並みの性能をもったノート型を商品を作れといっているところです．価格はデスクトップより高いですが，省スペース，低消費電力で有利になりますから，運転費用まで考慮するとデスクトップより安くなります．カラーTFT液晶は今のところ高価ですが，将来は安くなりますよ．NEC鹿児島に工場のラインを増やしますし，来年はNEC秋田にも専用工場を作ります．今後は，カラーTFT液晶の工場を持っているところが有利になります．

エプソンがPCIバスを採用しました

少し先走りしていると思います．まだあれが国際標準になるか分かりませんし，もしなったと仮定しても，あのボードの大きさが国際標準になるとは思いません．PCIバスはAT互換機のメーカーも採用するところが多くなってきていますが，そうなるとあのバス形状はどうなるでしょうか．

98でやれないことをやろうとしているようですが、リスクが多いです．自分で市場を開拓してマジョリティを持っているところがやるならばいいのですが．あえていうならマニアは買うかもしれません．ただマニアばかりにいい顔を見せても商売にはなりません．もちろんマニアの方にも買っていただかないといけませんが．



Pentiumプロセッサのアップグレードにしても，まだ石が出ていないのにアップグレードするのは，なんかおかしいと思っているのです．うちでもソケットを入れたときは，コスト高になるので反対だったのです．Intel社のチェックプログラムを通ってますから保証はされていますが．

富士通がついにAT互換機をだしましたね

富士通は，コンピュータメーカーとしてすごいことをやったと思いますよ．冒険ですね．これまでのFMRのお客様はどうするのでしょうね．事実NECへ問い合わせがきています．FM-TOWNSもそうです．FM/VとTOWNSとでは世界が違うのに，あれではお客様は違いが分からないですよ．うちが98MULTiを出すのとは違います．98MULTiはFELLOWとMATEと互換性がありますからね．

パソコンという同じ領域内でこういうことをやっていいのかなと思います．苦しかったんじゃないでしょうか．そうなると，先行きはその内のどれを選ぶかでしょう．コンピュータメーカーは継続性を大事にしてやっていますから，どれかひとつに絞らないと難しいでしょう．ということはFMRとTOWNSの継続性はあきらめたのだろうと思います．そして継続するのはFM/Vでしょう．富士通の社内でも混乱しているようですね．販売店さんにいたっては,今までFMRで勉強して育っているのに，ああいうことをしてはいけませんね．

OS/2 2.1やWindowsNTはどうなりますか

OS/2は累積3万3千本ぐらいでています．現在のバージョンは1.2ですが，IBMと共同でOS/2 2.Xを開発

しているところです．継続することが大事ですから．出荷日はまだいえません．

WindowsNTはマイクロソフト社と98用として開発中です．ネットワークサーバ用の大型OSですが，i486DX2/66MHzのMATEで動きますので，上位機種のMATEがサーバ機能を果たします．またNECのサーバ専用機でもWindowsNTでいきます．そしてクライアントとしては，Windows3.1とMATEの組み合わせです．出荷は1月末の予定です．過去のMS-DOSやWindowsの場合をみても分かりますように，OSは苦労しながら立ち上がってきています．WindowsNTも立ち上がりにそれなりに苦労すると思っています．すぐ花開くというわけにはいかないでしょう．それでも，NECは他社に先駆けてWindowsNTを出していきます．

マイクロソフト社と日本IBMはMS-DOS 6/Vをだしますが，NECではMS-DOS 6を出しますか

DOS6で出すのか，あるいは6はやめてDOS7になって出すのかは決めていません．ただDOSも継続しなければなりませんから，開発は進めています．まだ工場で検討している段階です．

やはり問題は互換性を維持できるかどうかですね．ハードディスクが進歩してコストダウンしています．圧縮ソフトだけじゃ弱いですしね．

プラグ&プレイへの対応はいつごろですか

当然，次のステップとしてシカゴもマイクロソフトと開発しています．これはOSメーカーと装置メーカーが協力してやらないとできないですから．

ただね，プラグ＆プレイなどと，かっこいいことといってますが，本当かなというのはあります．まだ



実体が見えませんので，良く分かりませんが．

パソコンにはアプリケーションがあるわけですが，ユーザの声を聞きますと，WindowsでDOSのアプリケーションをインストールするのは大変なことなのですね．まずこういうものを簡単にしなければ駄目だと思うのです．プラグ＆プレイよりもインストール＆プレイですよ．今は，一体型のパソコンが買いやすくなってきている状況ですから，そっちの方が大事だと思います．

モービルコンピュータが話題になっています．来年は普及にはずみがつくでしょうか

実際に使いものになるかが問題です．AppleがNewton Message Padをだしましたが，あまり売れていないようです．私もペン入力を使ってみたのですが，ペン入力処理はいらいらしますね．認識できませんというメッセージが出たり，認識されないならまだいいが，書いた文字とは違う文字が入力されます．予定表入れたりメモを書く程度なら100%認識してくれるペーパーの方がいいですよ．いらいらする装置は最悪ですね．装置というのは愕きと感動だと思います．期待感で買い，それに応えてくれることが必要です．

では，移動しながら使う装置というのは何でしょうか．それは読む装置ではないかと思うのです．と

いうことで実は今回デジタルブックを作ったんです．ページャ（ポケットベル）が今人気ありますが，あれは読む装置ですよね．メッセージを数字などで送ったりしてコミュニケーションをとっています．あの表示部分が大きくなったら多くのデータを読めますよね．たとえば文字放送です．これをデジタルブックで読めたら面白いですよね．ですから将来はモービルコンピューティングというよりモービルコミュニケーションツールというものが，伸びると思います．

まず，来年はこのデジタルブックがどう花咲くか期待しています．デジタルブック向きの内容はHOW TOものがいいですよ．将棋がいい例ですよ．ボタンを押すと次の一手の駒が順番に動いていくのです．ペーパーメディアでは体験できないことを体験できる．

98のサブノートはいつごろ出しますか

作れっていっていますが，まだのようです．ハンディ98で苦労しましたからね．サブノートはハンディ98のイメージがありますから開発する連中もみんな困っているようです．来年は出るでしょう．

エプソンがA5のサブノートをだしましたが，あれは高いし，中途半端ですね．ペン入力にこだわったのでしょう．見た目はかっこいいですが普及はしないでしょう．ハンディ98のようになったらまずいのではないでしょうか．

どちらかというとThinkPad 220のようなタイプがいいですよ．IBMの製品で光っているのは220でしょう．その次に期待されているのがPS/V Visionだといわてれいます．しかしそれはあくまでもメインの商品ではないと思います．サブノートはメインのビジネスにはなりません．やっぱりサブになりますよ．全体からみたら5割も売れないし，単価も安い．事業はトータルビジネスですからね．やはりハイエンドからサブノートまでラインナップをそろえてやらないとだめですよ．デジタルブックも含めて．そういった多様化をやっていかないと市場は広がらないですよ．

来年の抱負と戦略をお聞かせください

来年は，もっとパソコンが売れてほしいと考えています．そのためには市場を開拓する必要があります．ただ装置を出して，機能・性能のコストパフォーマンスアップや改良をやるだけではだめです．それ以外にもっと売る力とか，市場を開拓する力をつけるなどをしていきます．

たとえばPCカレッジ*は，今，秋葉原だけですがこれをもっと全国に広げていきます．またマルチメディア工房を作りましたが，ここから新しい作品を作り出し，新しいメッセージを伝え，新しい魅力を出していくことです．商品としては連続性を強化していきます．CD-ROMベースの98MULTiの強化も考えています．来年への期待は一太郎のWindows版がでるということです．

最後に来年に向けてのキーワードを？

「パソコンアップワードの時代」です．

来年は，大型コンピュータの方々がダウンサイジングを検討していますから，パソコンでそれをうけとめてやろうということです．つまりパソコンからみますとアップワードになります．来年はその環境が整ってきますよ．

◆

＊PCカレッジ（自動車教習所の単元と似たパソコン講座．1つの単元が終了すると次の単元に進むことができる）．



モービルの幕開け

キーマンに聞く－1994年の展望－

横河・ヒューレット・パッカード株式会社
コンピュータプロダクト営業本部
パーソナルインフォメーションプロダクツ
ジャパンマーケティングセンタセンタ長
八重樫　行男氏

100LXは95LXと比べるとかなり機能アップされていますね

八重樫氏（以下省略）:95LXと100LXは開発のスタートポイントが違うんですよ．我々の考えているモービルコンピューティングの世界は，MIA（Mobile Information Appliance）という意味でとらえ，この中を大きく三つのセグメントに分けて考えています．一つは，ニュートンや電子手帳に代表されるPDA（Peresonal Degital Assistant）の分野からのもの．一つは携帯電話の分野からきたPDC（Personal Degital Communication）の分野．そして，パソコンから降りてきたPBE（Personal Business Extension）の分野です．この三つの領域が重なり合っていて，これをどんな切り口で料理しようかというのが基本的なスタンスです．95LXのスタートはPDAから入っています．それに対し，100LXはPBEからの発想で，つまりDOS5の互換機を95LXと同じケースに押し込めたものです．ですから外観は同じですが，中身が全く違いますので，95LXを使っている方が100LXを使うとびっくりしますよ．世の中ではPDAがモービルだと言われてますが，われわれはPBEにおけるモービルに十分な価値を見いだしているわけです．

開発者が95LXのときにはできなかったことを100LXで実現させたと聞きましたが

たとえばアプリケーション間の連携プレイです．つまり，Windowsでいうカット＆ペーストの機能です．95LXではできないのです．逆な言い方をすると，DOSの環境にもかかわらず，100LXはメモのアプリケーションでカットして，1-2-3のところにペーストしたり，または電話帳に入れたりできます．そういう意味で100LXはDOSのPCを完全に超えているのですよ．

これはアプリケーションがROM化されているがために可能なことで，こちら側が連携プレイの機能を加工してつけることができるということです．

たとえばロータス1-2-3を開いている最中に，DOSに落ちて，DOSの中で何かファイルを見て，ちょっとファイルをいじってまた戻るということは，不可能ではないですが，通常環境では難しいです．でも100LXだと普通にできるんです．DOSマシンとしての使い勝手を最大限発揮したものです．あれがそのままデスクトップに載ったら，すごく使いやすいDOSマシンができるんじゃないかと思うぐらいですが．

OmniBookはどんなところにこだわって開発されたのでしょう

まずこだわるようですが，我々はサブノートとはいわずに，スーパーポータブルと呼んでいます．

われわれは，まず電池の持続時間をアメリカのビジネスマンがコースト・トゥー・コーストのフライトの中でフルに使えることを条件に開発しました．最低6時間は使いたいというところからスタートしました．さらに充電池がなくなったら，単3も使えるバッテリー形態にするということです．

軽くして，電池寿命を長くするにはどうするかというと，CPUを1～2ランク落として消費電力を落とし，そのかわりアプリケーション環境をROM化することで速さを維持しようと考えました．また，ディスクを仮に使ったとしても，アクセスを減らしていこうと．さらにフラッシュメモリを使用しているときは，フライトの中だけではなくて，家に着いてもまだ使いたいということで9時間という長い電池寿命を実現したのです．ハードディスク使用時で5時間です．重さは1.3kgが限界です．DOSだけなら100LXの大きさでよかったのですが，Windowsを使うとなるとこの大きさになりました．



今度発表したOmniBook425は，CPUにTI社の486SXLVを載せたもので，完全なマイナーチェンジです．時代は486ですから．486のフルパワーではありませんが随分速くなりました．ただし，電池の消耗は激しくなり，10%ぐらい落ちています．スーパーポータブルですから，デスクトップの流れを追いかけながら，筐体はそのまま維持していきます．日本ではOmni Book300とOmni Book425を来年1月同時出荷を予定しています．値段はOmni Book425はOmni Book300の値段をほぼキープし，Omni Book300を下げますから買いやすくなるかもしれません．

将来的にはカラー化もありえます．今後，速くなっても電池寿命も長くできる方策がわかったら，ROM化する必要はないので，ROMを外したものも考えています．

OmniBookは日本市場で受け入れられますか

米国ではモービルに対して，どういうニーズがあるかというと，データをつくる，あるいは加工する，あるいは絵をつくるまでいくのです．つまり，デスクトップにある環境をどこまで持ち出すかということです．例えばグラフィック機能を使うフリーランスやパワーポイントをモービルで使いたいよという人だっているわけですよ．

一方，日本でのニーズはほとんどワープロなのです．となると，テキストレベルでのニーズしかないところで勝負できるかといったら，厳しいですね．本当の価値が発揮されなくなりますから．

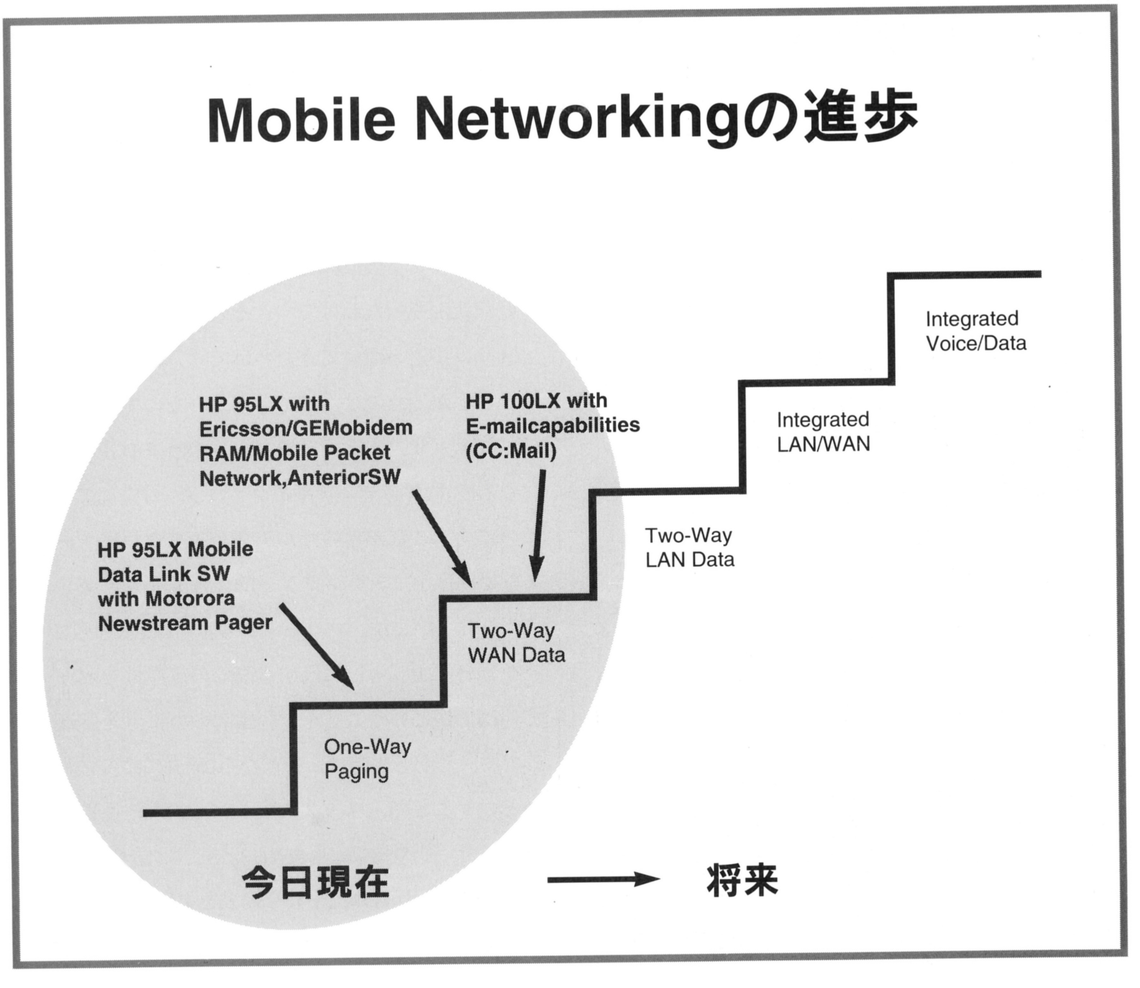
しかし今期待しているのは，先ほどコースト・トゥー・コーストと言いましたが，もう一つのフライトのイースト・トゥー・ウエスト，つまり東京とカリフォルニアのオーバー・パシフィック・オーシャンの方々が，その辺の価値を見出してくれるんじゃないかなと思います．

モービルコンピューティングの現在の状況はどのようなものですか

こんな図が世界の標準ステップだと考えています．ワンウェイページング（ポケットベル）から始まって，どちらかというとWAN(Wide Area Network)に広がり，将来はLANやインテグレーテッドなものになるでしょう．実は95LXは，ワンウエイページングには耐えられるモービルだったのです．RAMや，あるいはEMBARCとか，いろいろなワンウエイページングのサービスがアメリカではありますから，95LXにMOTOROLAと共同で作ったユニットをつけることで，ページャーになります．現実に米国には60万人の95LXのユーザがいますが，そのうち10万弱のユーザがワンウエイページングのMOTOROLA製ユニットをつけて，実際のビジネスで使っています．最近はニュートンが同じことを始めています．

また，Lotusのcc:MAILは，米国で400万～500万ユーザと言われ，とてつもないユーザがいるわけですよね．それがVANだったり，ゲートウエイサービスだったりして，さらにポスト・オフィスというのがあちこちに点在しています．そこに何らかの無線でアクセスすれば，あとはWANの環境でのMAIL環境が使えるのです．要はアクセスポイントがありますから，電子メールのコミュニケーションができるんです．

日本ではいわゆるVAN的な発想ではまだまだ普及していませんし，cc:MAILはきわめて限定されています





から，全然役に立たないです．

また，PCを持っていれば，コンピュサーブにごく普通に入っているわけです．そこにも当然ゲートウエイサービスがありますから，いわゆるWANという世界が構築されているのです．つまりモービルではLANよりも先のWANのレベルに来ています．

日本の環境が遅れているということですね

実は，モービルコンピューティングの普及の要となっているのはPCのLANなのです．LANが普及しているからこそモービルの価値があるのです．

ところが，日本のPCは，年間二百数十万台出ている中で，ネットワークにコネクトされているPCは，2割なくて十数%です．

モービルのメリットというと，コミュニケーションと即時性ですね．LANのメリットとしては，たとえば，あるプロジェクトのメンバーが，全部，電子メールやブリチンボードを使って，進捗やアイディアを張りつけると，いろんな人がいろんなコメントを返してくれます．いちいち，集まって何かやるのではなくて，思いつくまま何かやりつつプロジェクトを進め，プロジェクトマネージャーはそれを見ながら全体をマネージしていく．こういった仕事の進め方が行われています．しかし日本にはそういう慣習は十分育っていないわけです．

日本の経営者の方に問いたいのですが，いまの生産性に満足されていますか．その生産性を阻害している要因は何ですか．時間は正しく使われていますか．時間に対して正当な労働評価がなされていますか．そういうことを経営者が考えなければだめですよ．アメリカの三分の一だ，四分の一だと言われている生産性を，人を増やしたり時間を伸ばすことで何とかキャッチアップしています．しかし，これからは人を絞って，そのことでコストが下がり競合力がつくという時代がくると思うのです．

物を作ることに関しては日本では非常に進歩しました．それが80年代後半から90年代前半の日本のマニファクチャリングの成果であり，その結果としての経済の好転状況だったわけです．それがここへきて今度はアメリカが追いついてきたわけです．いま，くっきり浮かび上がったのはホワイトカラーなのです．たとえばプロセッサ，あるいはRISC，それからDSPなどキーになるLSIのアイディアやテクノロジーは全部アメリカなのです．これからのマルチメディアなり，キーになるDSP自体が，テクノロジー的には98%もアメリカに抑えられています．プロセッサも結局はIntel強しです．

そういう頭脳労働に属する部分の生産性をどうやって伸ばすかということを，経営者の方に考えてほしいのです．しかし当時は，それならR&Dに投資すればいいということで，ワークステーションへの投資が始まりました．その結果としてYHPとSUNは儲かりました．それでは，それ自体，十分な効果を上げたかというと，電気系のCADと機械系のCADのごく一部であって，本

当にクリエイティブな発想なり，クリエイティブなツールとしてのPCやワークステーションは，まだまだ普及していないわけです．そこに経営者のパラダイムがシフトしない限り，残念ながらLANはまだ普及しないでしょう．

日本で整っている通信環境は何かというと，ファックスネットが世界で一番発達しています．よく言えば，ドキュメントベースでのコミュニケーションが日本ではまずありきです．まずファックスが先に入った．それが一段落として，やはり再加工できないし，不便さが出てくる．ホワイトカラーの生産性が，いま低いと言われているのは，結局はファックスのせいかもしれないということに気がついたときに，やっと次のキャラクタベースのステップが出てくるかもしれません．

逆にアメリカは，再加工ができるキャラクタベースのコミュニケーションが進んでいますが，ではなぜ最終的にファックスが必要になったかというと，キャラタベースではドキュメントの美しさが保てないからです．グラフィックの要素や，文字のフォントの要素，美しさの用途を加えようと思ったら，ファックスになってしまう．

したがって，使われ方もPCのアプリケーションから直接ファックスを送ります．これが主流になっているわけです．

来年の展開は？

来年はLANのビジネスに力を入れていきます．またそれによりモービルの下地ができてくることになります．

モービルのもう一つの下地は，100LXが数多く出ることです．LANとモービルの両方から進めていくことで，ようやく1995年に何かできるんじゃないかと思っています．



「外出時は専用のホルダーを使い背広の下に100LXを携帯するという」



FM/Vでシェア30%

キーマンに聞く－1994年の展望－

富士通株式会社
パーソナルビジネス本部
副本部長/常務理事
神田 泰典 氏

AT互換機のFM/Vを出されました．FMRやTOWNSとの住み分けはどうなるのでしょう

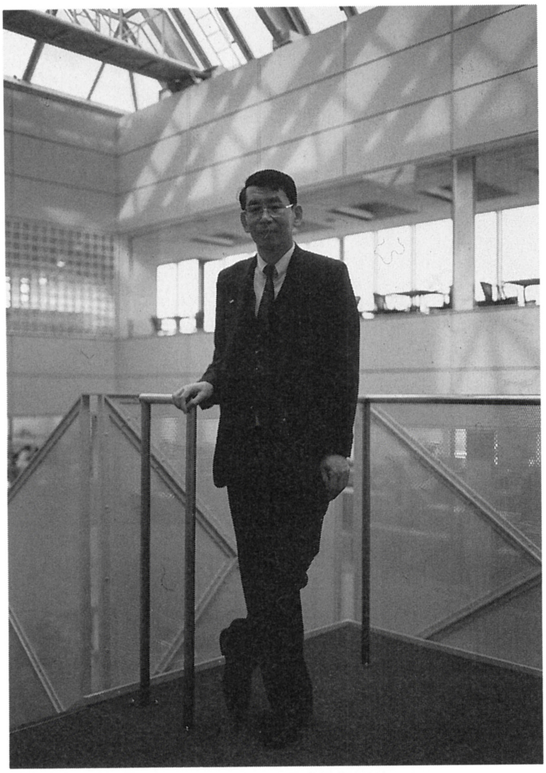
神田氏（以下省略）:いままでFMRとTOWNSをやっていたわけですが，AT互換機を出して，富士通も国際標準機でビジネスをやりますと発表したわけです．いろんな意味でその反響は大きかったですね．もしこの発表を2～3年前にやっていたら大騒ぎだろうと思いますが，今はユーザの方が時代の流れをよく知っていますから冷静です．逆にこのまま古い体制を維持するとかえって不自然です．やることが合理的だということで好感も得られているようです．

世界中でパソコンが1,000万台のレベルで作られている中で，日本市場だけは違うものを作ってきたわけです．コスト的にみても，数が作られる分，値段もそれだけ下がるわけですから，今後は価格的にも成り立たなくなります．またソフト資産をみても日本以外ではみな共通で使われているわけです．これからは日本だけが違うことをやっていても絶対うまくいかないですね．富士通も，海外のビジネスに対してはAT互換機をサポートしてきました．ですから長い目でみますと，FM/Vがメインになると考えています．

FMRは，富士通の大型システムをサポートするために作ってきたもので，通信やホストの端末として使われています．もちろんいままでのお客さんは，その上にソフト資産が多数載っていますので，ずっとサポートしていきます．

TOWNSの方はちょっと違っていて，これはマルチメディアに向けたマシンで，まだまだこれからやっていく分野がいろいろあると思います．例えば教育の分野などに特化したもので，新しいタイトルを作っていきます．TOWNSは簡単に使えますので，教育分野や一般消費者寄りを目指してやっていくつもりです．

実は，いままでTOWNSでやってきたことが，Windows3.1のマルチメディアの機能の中に大分取り込まれているのです．それはMultimedia PC Marketing Councilというコンソーシアムがあり，そこで仕様を決めているのですが，その仕様(MPC規格)は，先にやった人なり機械なりのアイデアをそこに入れてスペックを決めていくのです．TOWNSでやった仕様も，かなり参考にされて入ってきているわけです．そういうのを先進的にやっていくプラットフォームだと位置づけています．



OSの戦略はどう考えていますか

これまでUNIXとWindowsNTの戦いとか，OS/2とWindowsNTとどちらがいいかというのがありました．UNIXは，一時はホストコンピュータの後がまと言われていたのです．でもそれも多分ないでしょう．UNIXはオープンだと言われていますが，何とか連合がこちらがいいとか悪いとかやっているわけですよ．一般のユーザから見れば，ユーザ不在の話なんですね．基本的には同じだけど，結局みんなちょっとずつ違うわけですよね．

一方WindowsNTというのは，マイクロソフトが全部押さえていてソースは公開せず，バイナリだけで動かしているわけです．互換性が非常に強いかわり，ユーザが中をいじったりはできないわけです．

大学だとか研究所だとかで使うのなら，UNIXがいいと思うし，これからも使われ続けると思います．

WindowsNTも，機能的にみると中身はUNIXみたいなものです．商品としてのコンセプトなりパッケージングが違うというだけです．それはビル・ゲイツの商売がうまいと言えば，ただそれだけの話かもしれません．

独占はよくないのでしょうが，ある程度統一されたものがあってはじめて，コンピュータを使って何かするという目的が達成されるわけです．そういう意味では，WindowsNTで落ちついてきて，業界自身は伸びていくのではないかと期待しているのです．

また，OS/2ですが，端末のOSとして日本では大分入っています．ただ，OS/2はIBMが作っていますから，サポートの問題とかになると，富士通がやってもなかなかうまくいかないのでは，という心配があるわけです．IBMはやはりコンピュータメーカーですからね．ところが，OS/2とWindowsNTが競合するようになって，結局はWindowsNTが勝ったということです．そういう意味で，OS/2は多分これからはそんなに伸びないと思います．いまOS/2を使っている人も，これから先はWindowsNTへどうやって移るかを考えようというところです．

パソコンを中心にしていこうという立場から見ると，マイクロソフトとIntelの力がメジャーになり，両者がパソコンの世界でリーダーシップを取ることが，はっきりしてきました．ですから，これから先はWindowsの一本道なのですね．これまでのように選択したり両方ともやる必要はあまりないのです．営業的にも，技術的にも，開発でも，その道の方向へ進めばいいわけです．そういう意味では非常にやりやすいとも言えます．UNIXの方がいいかとか，UNIXもどれがいいかとか，そんなことを言っていると，全然仕事が手につかないわけですよ．

OSを一社に独占されることの危険性はないですか

それも考えようです．いまは統一したものが必要だ

と思うのです．富士通から見るとマイクロソフトはソフトを作っているだけですから，パートナーなのです．IntelはCPUを作っていますが，日本で富士通とはぶつかっていないわけです．そういう意味では，昔のIBMは，大型汎用機分野でマイクロソフトよりもっと独占的でした．IBMはソフトとハードも作っていましたから，富士通と完全にコンペチターでしたね．ご存じの通り，当時はIBM互換機を作るにしても，外部仕様だけで作っていたわけです．

また，「Windowsなんかに一本化されるのはけしからん，日本人も何か考えろ」という話もあるかもしれません．しかしMS-DOSと違うOSを作ったところでどうってことありませんし，Windowsと違うOSを作っても，どうってことありません．本来は，私たちがその上で何を作るかということの方が大切だと思うのです．

産業の進歩により部品が発達し，部品から製品を作るようになると，その部品は共通化されるわけです．パソコンにとって，CPUは部品ですし，OSも部品です．WindowsNTも部品なのです．その部品を使って何を作るかということが大事です．

私が入社したころは，アーキテクチャも自分で考え，回路も設計し，トランジスタで回路を作り，コンピュータを作っていたわけです．そのときは，各社が互換性のないコンピュータを作っていたわけです．

重要なのは，部品を作るよりは，例えば一般の人とパソコンとの間に，まだギャップがある現状ですから，この間をどういうふうに埋めるかということに力を注ぐべきだと思っています．

オアシスにモデムを内蔵するなど，ワープロ通信に力を入れているようですね

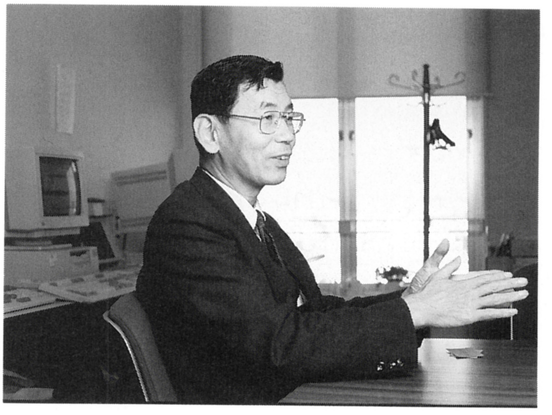
ワープロ専用機の使われ方をみますと，企業では手紙を書くために使っているところがとても多かったわけです．ワープロから印刷出力したものを封筒に入れ郵便で送っていました．これがモデムを使って直接通信で送れるようになれば，プリンタも紙も切手も封筒もいらなくなるのです．

私は将来は郵便がなくなると思っているのです．郵便は時間がかかりますし，料金は高いですね．あんなに人手かけて，大の男がスクーターに乗ってくるわけですよ．それで62円とかでしょう．まずコスト的に引き合わなくなります．そのうちまた郵便料金は上がりますよ．それに郵便は絶対時間がかかります．その点，通信ですと日本でもアメリカにいてもすぐに届きます．それにすごく楽ですよ．返事もすぐ出せますしね．電話よりもすごいところは，電話はその人の電話番号にしかかからないのでそこにいなければダメなのですが，電子メールは，その人のIDに送りますからどこにいようと関係ないのです．もちろん，必ずアクセスする作業が必要ですが，アクセスさえしてくれる相手なら必ず届きます．

手紙は，東京都内の速達でもすぐには届きませんからね．ファックスはどこへ行くか分かりませんし，番号を間違えると危ないです．通信は特にビジネスで使うにはすごくいいですね．コストも手間もかからない．

将来，郵便がなくなる代わりに，NIFTY-Serveなどのネットワークの会員数が1,000万人まで増えていきますよ．

なぜNIFTY-Serveの会員数が伸びてきたかというと，やはり人間は，情報よりはコミュニケーションということです．データベースにアクセスするという要素もあるのですが，一般的にみんなが使っているのはコミ



ュニケーションです．全然違う分野の人とすごくよく知るチャンスがあるんですよね．電話だとそういう機能はないですよね．

ネットワークの性格もアメリカと日本とでは違うと思うのです．例えばCompuserveの会員は，パソコンのユーザが大部分なのです．NIFTY-Serveではパソコンユーザもいれば，ワープロ専用機ユーザもいて，かなり幅広い人たちが入っています．アメリカでは，階層社会になっていますから，ネットワークに違う階級の人はあまり入ってこないと思うのです．要するに，一つのネットワークは用途別に限られているのです．そのかわりネットワークの数が多数あるわけです．

日本は朝日新聞とか，毎日新聞とか，読売新聞といった全国紙が何百万部も売れているでしょう．アメリカはローカル紙ですからそんなに部数は出てないのです．新聞と同じように使われ方が全然違うと思います．ですから，ネットワークの会員数もアメリカではそんなに多くならなくても，日本ではまだまだ伸びると思います．1995年で100万人で，2000年で1,000万人というのをNIFTY-Serveの目標にしているのです．

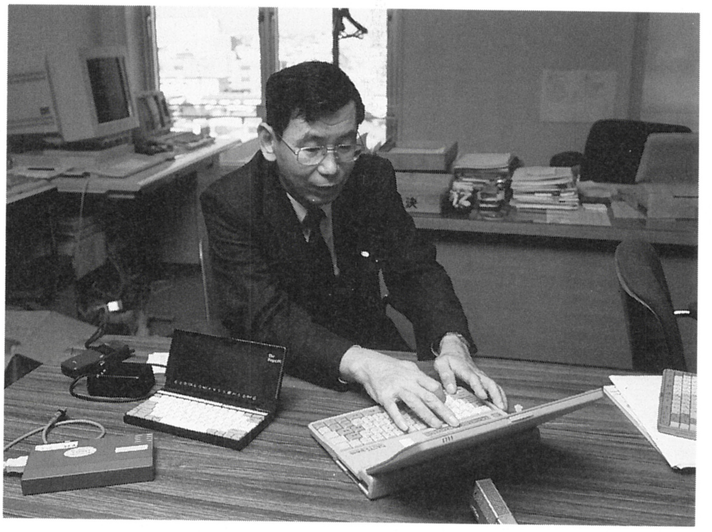
ワープロ専用機だって現在は1,000万人のユーザ数はいっていますよ．初めてワープロ専用機を作った当時は「日本人がみんな使うようになるよ」と言ったら，「えっー，神田さんの話は面白いね」なんて笑っていたのですからね．

出荷台数は，ワープロ専用機だけで年間250万台くらいですから，パソコンより多いのです．何といっても使っているユーザの層が広いですから．でもユーザにとってはそんなのはどうでもいい話で，そのうちパソコンもワープロ専用機も操作性がよくなってどっちでもよくなりますよ．

操作性といえば，プラグ&プレイが提唱されていますね

マイクロソフトは去年あたりまでは，いままでパソコンでできなかったことをやろうというので，それこそペンコンピュータだとかマルチメディアの時代とい





ってましたが，この2年や3年ではどうなるものではないことが分かってきたのでしょう．落ちついてきました．そこで，いまの使い方でもっと使う人を増やそうということで，プラグ&プレイがでてきたのですね．

これまではパソコンにカードを挿しても設定してあげないと動かないわけです．ところが設定ができる人は限られているわけです．結局，そういうのがネックになるとどんなにパソコンが安くなっても，お客さんが使えないわけです．

その点，ワープロ専用機ならプラグ&プレイどころか，買ってきたらパソコンということを意識しないですぐに使えるわけですからね．結局，パソコンもそういう方向に向かっているという一つの動きですね．

本当は一体型で家電商品みたいにした方がいいのです．自分で組み立てたり，カードを抜き挿しするものではありませんからね．これからはそういう方向へ行きますよ．アメリカでワープロ専用機がないのが不思議ですね．アメリカで作ったら売れるんじゃないかと言っているのですが．

来年へ向けての抱負を

まず全体の話をしますと，富士通はご存じのように，何十年も前から，大型コンピュータをやってきたわけです．これはIBMが開いてきた世界なのです．日本が情報処理産業で伸びてきたのも，大型がベースでした．しかしアメリカではすでにパソコンの占める比率が高く，半分ぐらいがパソコンと言われていますが，日本はまだホスト中心で来ているわけです．これからは日本でも，大型コンピュータよりはクライアント・サーバーへいくのが時代の流れでしょうね．それで1993年はパソコンの世界になってきているのがはっきりと分かった年ですね．1994年になれば，坂を転がっていくようにその方向へ向かうだろうと思います．

具体的には，現在の富士通のパソコン市場シェアは12%くらいですが，来年はFM/Vに力をいれていき，1996年には30%のシェアの実現を目標にしていきます．富士通は大型汎用機も含めた全コンピュータのシェアは大きいですから，逆にいうと30%くらい取らないと，このダウンサイジングの流れにのれず，今までのお客さんを失うことになります．この半年が勝負だと考えています．もちろんPCIやPentiumプロセッサ，CD-ROMも考えていますし，WindowsNTは我々のメインの仕事になるでしょう．この1年くらいでパソコンの地位がだいぶ変わってきていますから，来年はパソコンがサーバーとして働くようになっているでしょう．



Geo Worksが動いているThe Poqet PC

特集

サブノートの気になるあのこと



■空調装置付きの部屋を必要としたコンピュータは，いつしか机の上に載るサイズとなり，持ち歩けるもの，手のひらサイズのものまで登場するほど進化した．サイズだけが原因ではないが，コンピュータがかなり身近な存在になったことは確かだ．
■そんな中，最近，急に耳にするようになった『サブノート』という単語．どうやら従来のノートパソコンより小さなサイズのものであることは分かるのだが，いったい何をしてサブノートなのか，その定義は何なのか，誰のためのものなのか，どんなことに使うのか，何が求められているのか，そんな疑問も湧いてくる．
■サブノートという新語には，期待感と不安が混在しているようにも見える．そこで，本誌では，世間一般でサブノートと呼ばれるものを集め，その実力と使い勝手を探ってみた．さらに各パソコン誌で活躍中の方々にご意見を拝聴し，サブノートの真実を探ってみることにした．（法林岳之）

❶ サブノートの真実
～PCユーザへのアンケート結果から考える“サブノート”って何?～

❷ どこまで使えるサブノート!?
～サブノートの実用性と問題点を考える～

❸ PCMCIAの周辺

❹ 香港ABCコンピュータ，リチャード・ロバート氏に聞く

❺ サブノートPCレビュー

❻ COMDEX / Fall '93 PCMCIAプロダクツレポート

1

サブノートの真実

～PCユーザへのアンケート結果から考える“サブノート”って何？～

●著者●
法林岳之
Kenichi JAWAMURA

●概要●
日本アイ・ビーエムのThinkPad 220の登場を皮切りに，ちょっとした『サブノート』ブームが起こりつつある．
コンピュータ関連各誌でもすでに「サブノート特集」が組まれているが，
実際に「サブノート」と称するマシンを見てみると，
ひとくくりにはできないほど，外形寸法も異なれば，重量も大きく異なる．
そこで本稿では，本誌執筆陣ほかに協力をお願いし，
現在の「サブノート」がどのように捉えられているのか，
普及のためのカギは何か，をまとめてみた．



サブノートとは何だ?

コンパクトなマシンを語るとき，まず忘れてはならないのが98ノート，ダイナブックによって切り開かれたノートパソコンというジャンルだ．それまでの携帯型マシンは『ラップトップ』という言葉で呼ばれていたが，とても膝の上に載るようなものではなかった．そんな状況の中で登場したノートパソコンは，約3kgの片手で持てるマシン，電源がない場所でも使える本格的な携帯用マシンというキャッチコピーとともに，ユーザの支持を得た．また，価格的にも安いことから，初心者用の入門機としても安定した人気を得るようになった．

しかし，携帯性を売りにしていたノートパソコンも，実は省スペース性を重視しているユーザが多いことが注目され，いつしかカラー液晶を搭載，CPUは386以上，ハードディスクもパック式が採用され，少しずつ重量級のものが増えてしまった．

時は'93年，ノートパソコンの主戦場がカラー液晶マシンへと移行し，DOS/Vの普及によってIBM PC互換のノートパソコンも数多く販売されるようになった．そこへ登場したのが『サブノート』である．この『サブ』という単語が，それまでのノートパソコンよりも小さいという意味なのか，デスクトップ機を持つユーザが補助的に使うマシンなのかは不明だが，いずれにせよ，ノートパソコンよりも小さいマシンを指していることは確かだ．サブノートという言葉といっしょに登場したのが『モービルコンピューティング』，つまり携帯型移動端末という概念だ．移動中にもコンピュータを携帯し，いつでも携帯電話や公衆電話からホストにアクセスできるという活用方法が提示されている．

サブノートという言葉も最も身近な存在に押し上げたのは，やはり日本アイ・ビー・エムのThinkPad 220

を置いてほかにないだろう．A5サイズで1kgというコンパクトなマシンは各方面で話題を呼び，いわばサブノートのお手本的な存在としても扱われるようになった．その後，いくつかのメーカーからサブノートが登場し，現在に至るわけだが，編集部や筆者陣との間でサブノートの企画が持ち上がったとき，「どこまでがサブノートなのか」という議論が持ち上がった．重さ，サイズ，バッテリー駆動時間，価格といった基本的なスペックをはじめ，メモリやハードディスクの容量まで，多岐に渡って議論が交わされたのだが，結論めいたものは導き出すことはできなかった．

そこで，ざべ流の『サブノートの定義』を見つけるため，サブノートのあるべき姿を探るため，各パソコン誌で活躍中のライター諸氏や著名人の方々にアンケート(以下，著名人アンケート)をお願いした．日頃からパソコンをよく触り，メーカーなどの動向などについても詳しい彼らの意見を聞けば，何らかの答えが見つかるかもしれないからだ．さらに，その補足といっては失礼かもしれないが，秋葉原でも何人かに同様のアンケートを実施した．詳しくは後述するが，予想以上にサブノートに対する認識が違ったことで編集部共々驚かされた．

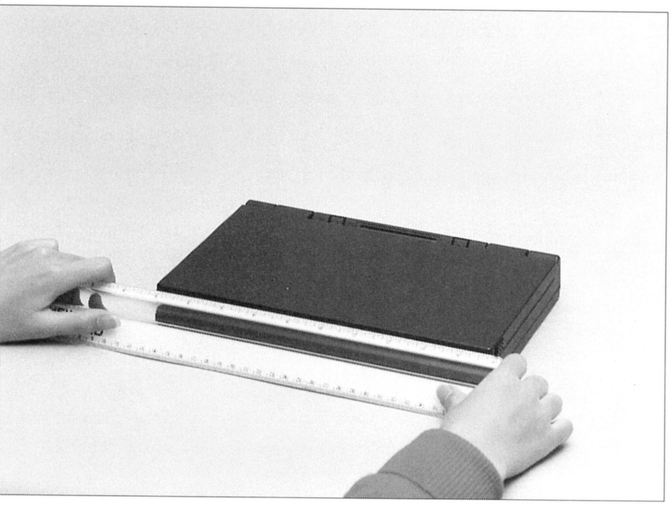
さて，ざべ流サブノートの定義では，どんな姿を描き出すのだろうか．

重さとサイズはどれくらい?

サブノートの姿を描くとき，やはり最初に出てくるのはThinkPad 220だ．アンケートでも，サブノートという言葉で連想するマシンとして必ず挙げられている．それだけインパクトが大きかったわけだ．

そのThinkPad 220の最大のセールスポイントは，やは





り1kgという重量だろう．学生やサラリーマンが持ち歩く鞄などの重さと，さほどの差もない手頃な重さといえるだろう．著名人アンケートでは，ほとんどの方々が1～1.5kgをひとつの目安と述べている．軽いに越したことはないが，やはりこの程度が限界なのだろう．唯一，今回のレビュー原稿でも執筆した**佐伯英和**氏が，

「FDDやACアダプタを含めて1.5kgなら許せる．つまり，本体だけなら1.2kg程度」

というコメントを残している．サブノートの形態を見ると，やはりボディをコンパクトに収めるため，FDDについてはすべて外付けになっているが，その重量も含めるという考え方だ．また，内蔵バッテリーが切れたときのため，ACアダプタの重量も加え，総計1.5kgに収めてほしいということだ．

たしかに，佐伯氏の言うように，サブノートでは装備するものも含めた総合的な環境を重量として捉えるのが得策だろう．バッテリー駆動時間の長いモデルなら，ACアダプタの必要性は少なくなるが，外出先でファイルのやり取りをするのであれば，FDDなどの周辺機器は欠かせない．ICカードでのやり取り，通信でのやり取りという方法もあるが，どこにでもあるメディアということになると，フロッピーディスクに何歩も譲ることになる．もし，携帯端末として使うのであれば，FDDやACアダプタ以外にFAXモデムやケーブル類の重量も考慮に入れるべきだろう．つまり，サブノートそのものの重量だけでなく，『サブノート環境』ともいえる状態の重量を考えたいということだ．

重さとも密接な関係にある『サイズ』についてはどうだろうか．これも大半の人がA5サイズ，もしくはA5とB5の間と答えている．A5サイズとB5サイズの間ということは，ノートパソコン全盛期に流行った言い方を借りれば，『A5ファイルサイズ』ということになるだろうか．つまり，A5ファイルサイズまではサブノートとして認知できるということになる．しかし，サイズの件については，実用に差し支えないキーボードが必要だという注釈を何人かが付け加えている．『月刊アスキー』『アプリンク』『NetWorks』の編集長を兼任する**遠藤諭**氏は，

「サイズはA5サイズ以下．キーボードはZEOS Pocketで十分タッチタイプ可能です」

としている．遠藤氏は知る人ぞ知るポケットマシン大好き人間だが，最近ではZEOS Pocketでメモを取る機会が多いという．ZEOS Pocketは富士通のOASYS Pocketにも似たマシンで，キートップのサイズも同じくらいのものだ．一方，日経MIXなどで活躍中の**内田暁(junk.35)**氏は，

「携帯性とキーボードの快適性を総合すると，B5程度がいいのではないか」

というコメントを残している．さらに，FAXモデムの記事などでおなじみの**びn**氏は，

「A5サイズでキーボードはできるだけ大きく」

これらのコメントは一見バラバラのようだが，実はサイズだけでなく，キーボードがタイプしやすいかどうかも重要なポイントだという点で一致している．どんなに小さくキーボードが使いにくくてはだめだし，多少は大きくてもキーボードが使いやすければ，十分実用になるという意味だ．

バッテリーの駆動時間は?

サブノートを携帯して使うとなると，気になるのがバッテリーでの駆動時間だ．著名人アンケートでは半数以上の人が3時間と答えていたが，もっと必要だという意見も多かった．いくつか例を挙げてみよう．まず，今回の特集で「PCMCIAの周辺」を執筆した**アホの坂田**氏は本人がHP-100LXを普段から使ってる関係か，

「最低でも24時間は動いてほしい」

と述べている．前述の遠藤氏もHP100LXを使っているそうだが，やはり，

「8時間はちゃんと動いてほしい．HP100LXやZEOS Pocketのように何日も動かないとストレスがたまる」

としている．FMR-CARDなどを使い，本誌でもおなじみの**宇野沢成夫**氏は，

「単3アルカリ電池2本で8時間以上．あるいは4本で10時間以上」

という条件を出している．3人ともHP100LXやFMR-CARDなどのバッテリー駆動時間の長いマシンを使っているためか，条件はかなり厳しい．どうやら，一度長持ちする環境を使ってしまうと，それ以上，短いバッテリー駆動時間では使えなくなってしまうのだろう．

しかし，考えてみれば，半数近くの人がなぜ3時間という数字を挙げているのだろうか．これについては，『スーパーアスキー』誌などでおなじみの**元麻布春男**氏が非常に明快な回答をしているので，その一部を引用してみよう．

「日本国内の移動については，3時間がある種のマジックナンバーであると思われる．新幹線で3時間という距離を東京を起点に考えると，神戸や盛岡が十分レンジに入る．それより遠方には航空機を使う方が一般的であり，航空機を利用して3時間ということなら，ほぼ日本全土をカバーできる」

なかなか説得力のある解説だ．これらの説を総合すると，最低でも3時間は連続稼働できることが条件で，予備のバッテリーを使ったり，バックライトを消す，使わないときはすぐにサスペンドするなど，ユーザ自身が工夫することによって8時間近く持ってほしいということに落ち着きそうだ．

98NS/E以降の98NOTEシリーズでは，セカンドバッテリーを持つことで3時間以上の駆動が可能になってい



たが，これくらいのスペックはサブノートでも満たしたいところだ．

CPUやメモリ，ハードディスクはどれくらい?

さて，CPUについてはどうだろうか．答えを先に言ってしまえば，用途次第ということになるのだが，著名人アンケートによれば，やはり486SX以上のものが求められているようだ．用途次第というのは，OSとして何が動くか，どんなソフトを使うかによって違うということだ．これについては，本誌「DOS/V Paradise」の連載でおなじみの**西川和久**氏が，

「DOSマシンなら386SL/20MHz程度で十分．Windowsマシンなら486SX/16MHz以上はほしい」

という区分を提案している．たしかに，デスクトップ機でもWindowsを使うなら486といわれるご時世．それなら，これくらいのスペックは当然だろう．386SX相当でもWindowsは動かないことはないが，仮に動いたとしても，それは「動いた」だけであって，「実用的」というにはほど遠くなる．ならば，あっさりと486SX(SL，もしくはSLエンハンス)相当のCPUを搭載してしまった方がいいのではないだろうか．

そう考えると，メモリやハードディスクについても答えは簡単で，Windows環境をサブノートで実現するならメモリ8MB，ハードディスクは最低でも40MB，現実的には80MB程度が必要になる．一方，DOS環境を重視するのであれば，メモリは2MB，ハードディスクも40MB程度で十分ということになる．



しかし，用途には，どんな人が使うかということにも密接に関係してくるだろう．そこで「どのような人がサブノートを使ってると思いますか」という質問をアンケートに盛り込んでみたところ，やはり編集者やライターが多いという答えが返ってきた．本誌「電脳騒乱節」「電脳曼陀羅」でおなじみの**中村正三郎**氏の，

「好き者」

という答えには笑ってしまった．ほかにも，

「自分の環境を常に持ち歩きたいマニアのためのものでしょう」(ライター・**戌野速太朗**氏)

「パソコン通信ユーザが中心かな」(東映無線・**鈴木仁**氏)

「おたっきーな人たちでしょうね」(ウェブテクノロジ・**小高輝真**氏)

といった評価が多く，ごく一部のユーザのオモチャいう見方が強い．しかし，業界にはマニアが使うことで注目度が高まり，一般ユーザにも普及するというセオリーがあるので，いずれは外回りの多い営業マンなど，一般ユーザにも使われることになるだろう．その時点で必要とされる用途がはっきりとし，CPUやメモリ，ハードディスクの容量なども決まってくるのではないだろうか．

サブノート普及のカギを握るもの

ここまで取り上げてきた要素すべてと密接な関係にある最後の砦が価格だ．現在，販売されているサブノートはいずれも20～30万円の価格が付けられている．しかし，著名人アンケートでは実売で20万円以下という指針が出ている．それもここまでの条件を満たした上でというあたりがなかなか手厳しい．前述の内田氏によれば，

「ブームといえるほどの売り上げを期待するのなら，実用最低スペックを満たしたマシンの実売価格が10万円を切ることが必要」

としている．10万円以下に落ちるにはコスト的にクリアすべき点が多そうだが，たしかに「サブ」という名にふさわしい価格になる必要はある．デスクトップ

これが『ざべ流サブノート』の定義だ

CPU	i486SX/25MHz相当以上(DOS環境のみなら，i386SL/20MHzでも可)
メモリ	8MB以上(DOS環境のみなら，2MB以上)
ハードディスク	80MB以上(MS-DOS環境のみなら，40MB以上)
サイズ	A5ファイルサイズ(約297×210mm．キーボードは両手でタイプ可能なもの)
バッテリー駆動時間	3時間(予備バッテリーなどで8時間駆動可能)
重量	1〜1.5kg
価格	20万円以下(実売価格)

機の中心モデル(i486DX2/66MHz搭載)が実売で30万円を切っているのであれば，サブになるものは2/3近い価格帯がポイントになるのではないだろうか．つまり，20万円が売れるものと売れないもののボーダーラインになりそうな気配だ．

一方，秋葉原での一般アンケートでは10〜15万円，15〜20万円と答えた人がやく4割ずついた．若干の差はあるが，やはり20万円を切ることがひとつめのステップで，15万円が最後の大きな壁になりそうだ．AV機器と同じ次元で語るべきではないのかもしれないが，ハンディカムのようなものがアクセサリーキットなどを含めて15万円を切ると，かなり売れ行きが違うという現状を考えれば，15万円という価格には意外に大きな壁なのかもしれない．

ところで，スペックや価格など，すべてを含めたものの中で，いったいどんなキーワードが普及へのカギを握っているのだろうか．著名人アンケートの中では予想通り，重量とバッテリー駆動時間という意見が圧倒的に多かった．いわばサブノートの生命線ともいえるものだけに，注目度は高いようだ．

そんな中で，「旅先通信レポート」(『月刊アスキー』連載)でも知られる**辻バード**氏は，非常に現実的なキーを提示してくれた．

「やはり，NECが98環境で，すごいマシンを出すことだと思う」

現在，市場に出ているマシンは，いずれもIBM PC互換機のサブノートばかりだ．98環境では'93年11月にエプソンが『PC-486Portable』を発表し，ようやくサブノートが登場したという感が強い．良くも悪くも98天国である日本の市場を考えれば，98環境でのサブノートがエプソンとNECの両社から発表されることが望ましい．市場では，NECが1月，もしくは2月に発表するといわれるサブノートへの期待が高まっているが，このマシンのでき次第で，今後のサブノート像も大きく変わりそうな気配もある．

さて，著名人アンケートと秋葉原での街頭アンケートをもとに，ざべ流『サブノートの定義』を探ってきたわけだが，そこで描き出されたサブノートは予想通り，すぐにでも飛びつきたくなるような代物となった．逆にいえば，それだけ現在のサブノートへの不満も大きいという見方もできるだろう．スペックを簡単に表にまとめておいた．

この定義に当てはまるマシンとなると，なかなか見つからない．ThinkPad 220も遠くはないのだが，現状ではCPUパワーとバッテリー駆動時間のマイナス点は大きい．今後，登場するマシンは，ハードウェア的なニーズはある程度満たせるかもしれないが，最後の最後に壁になるのは，やはり価格だろうか．いずれにせよ，中途半端に妥協をしないマシンが登場してくることを期待したい．

●

最後に，アンケートにご協力いただいた各氏には，この場を借りて深く感謝の意を表したい．誌面の都合上，すべての意見を掲載できなかったが，十分参考にさせていただいた．

また，ざべ流と名乗りながら，本誌以外での活躍が目立つライターの方や他誌の編集長にまでご協力いただいたのは，恐縮の限りである．しかし，そんなことも気にせずアンケートに答えていただいた各氏に力いっぱい感謝したい．

〈アンケートにご協力いただいた方〉
アホの坂田，一ヶ谷兼乃，内田 暁，宇野沢成夫，遠藤 諭，こうのたけし，小高輝真，斉藤重喜，佐伯英和，鈴木 仁，田中稔之，辻バード，中村正三郎，西川和久，びn，戊野速太朗，本田未智，元麻布春男
(敬称略・五十音順)
このほかにも，秋葉原で多くの方にアンケートにご回答いただきました．ご協力，ありがとうございました．(編集部)

特集・サブノートの気になるあのこと

2

どこまで使えるサブノート!?

～サブノートの実用性と問題点を考える～

●著者●

法林岳之

Kenichi JAWAMURA

●概要●

電子手帳よりもパソコンらしく，ノートパソコンよりコンパクト．
そんな触れ込みで急激に注目を浴びはじめたサブノートだが，果たしてどれくらいの実用性があるのだろうか．
どんなことに使うと効果的なのだろうか．
実用性を損なう点があるとすれば，どんな点なのだろうか．
いま，注目のサブノート環境を現実的な面から捉えてみた．



ノートパソコンが流行りはじめたとき，誰もがデスクトップ機で使っている環境をそのまま持ち込めると考えていた．もちろん，当時販売されていたラップトップパソコンと違い，バッテリー駆動ができ，重量も3kg程度と軽かったため，どこでもパソコンを使えるのではないかと考えた．

しかし，その後のノートパソコンの成長ぶりを見ると，「どこでもパソコン」は幻想でしかなかったことに気づく．なぜなら，その後に発売されたノートパソコンは，携帯性という特徴以外の部分に技術が注ぎ込まれたからだ．まずCPUを速くし，次にハードディスクを内蔵．メモリもノーウェイトのものが搭載できるようになり，液晶ディスプレイもカラーのものが増えてきた．値段も少しずつ高くなってしまい，誰もが手頃に買えるセカンドマシンという印象はなくなった．いまやデスクトップ機を上回る価格のものもあり，立派なファーストマシンとして君臨している．

こうした流れの中でサブノートパソコンが生まれてきたのは，やはり必然だったという気がする．つまり，ユーザは従来のノートパソコンが目指していた携帯性を重視したマシン，あるいは気軽に買える2台目のマシンを求めていたのだ．デスクトップ機は98とAT互換機があり，最新鋭のPentiumプロセッサマシンが50万円程度で買えてしまう．Windowsの登場により，アーキテクチャの壁もなくなりつつある．ならば，次に買うのは，持って歩けるマシンというのもうなずける話だ．“モービルコンピューティング”というキャッチフレーズも消費者の購買意欲をくすぐってくれる．マシンを携帯するという幻想がいよいよ現実のものになるのだ．

しかし，現在販売されているサブノートには，いくつかの問題点も指摘されている．前章で探った『ざべ流サブノートの定義』と実際のマシンでは，微妙に差がある．この章ではもう少し突っ込んだ形で，現在のサブノートの実用的な面と問題点について検証してみよう．

「何に使うのか」ですべてが決まる

さて，サブノートの実用性を語る前に考えておきたいことがある．それはサブノートの用途だ．サブノートを使う目的がハッキリしていなければ，どんなスペ



ックが必要なのかが分からなくなる．具体的にいえば，ざべ流サブノートの定義でも触れたように，DOS環境で使うのか，Windows環境を使うのかによって，CPUパワーからメモリやハードディスクの容量まで大きく違ってきてしまう．もちろん，OSだけでなく，アプリケーションソフトを何にするのかでも大きく違ってくるだろう．まず，サブノートの何のために使うのかをハッキリさせることが第一である．

例えば，筆者自身だけでなく，出版に携わる者は外出先でメモを取ったり，原稿を書くことが多い．原稿の締め切りに遅れてしまい，編集部に缶詰になるという情けない話もあるが，外出先に自分が普段使ってる環境と同じものがある可能性は低い．ワープロやエディタも違えば，日本語入力FEPも違うこともある．あるいは，辞書の鍛え方が違うために，思うように原稿が書けないということもあり得る．そんなとき，サブノートを持ち歩いていれば，他人のマシンを借りなくても済むし，自分の環境を他人のマシンに一時的に持ち込むことも不可能ではない．

また，外回りの多い営業マンが，Excelなどで作った見積り表を顧客に見せるというシチュエーションにもサブノートは役立つ．あるいは，プレゼンテーションに使うという方法もある．さらに，パソコン通信ヘビーユーザなら，最近街角に増えてきたISDN電話からホストコンピュータにアクセスするということも考えられるだろう．

いずれの場合にもサブノートは便利だが，すべてアプリケーションソフトが違うため，動作条件が違ってくる．Excelやプレゼンテーション（Lotus Freelanceあたりか?）なら，Windows環境が前提となるし，通信やエディタだけなら，DOS環境でも申し分ない．

こういった目的を絞り込むことによって，自分がサブノートに何を求めているか，どんなスペックが必要なのかは自ずと決まってくる．もちろん，それに応じて，現在のサブノートに何が欠けているか，何がムダかもハッキリしてくるはずだ．

重量とバッテリー
～駆動時間とのジレンマ～

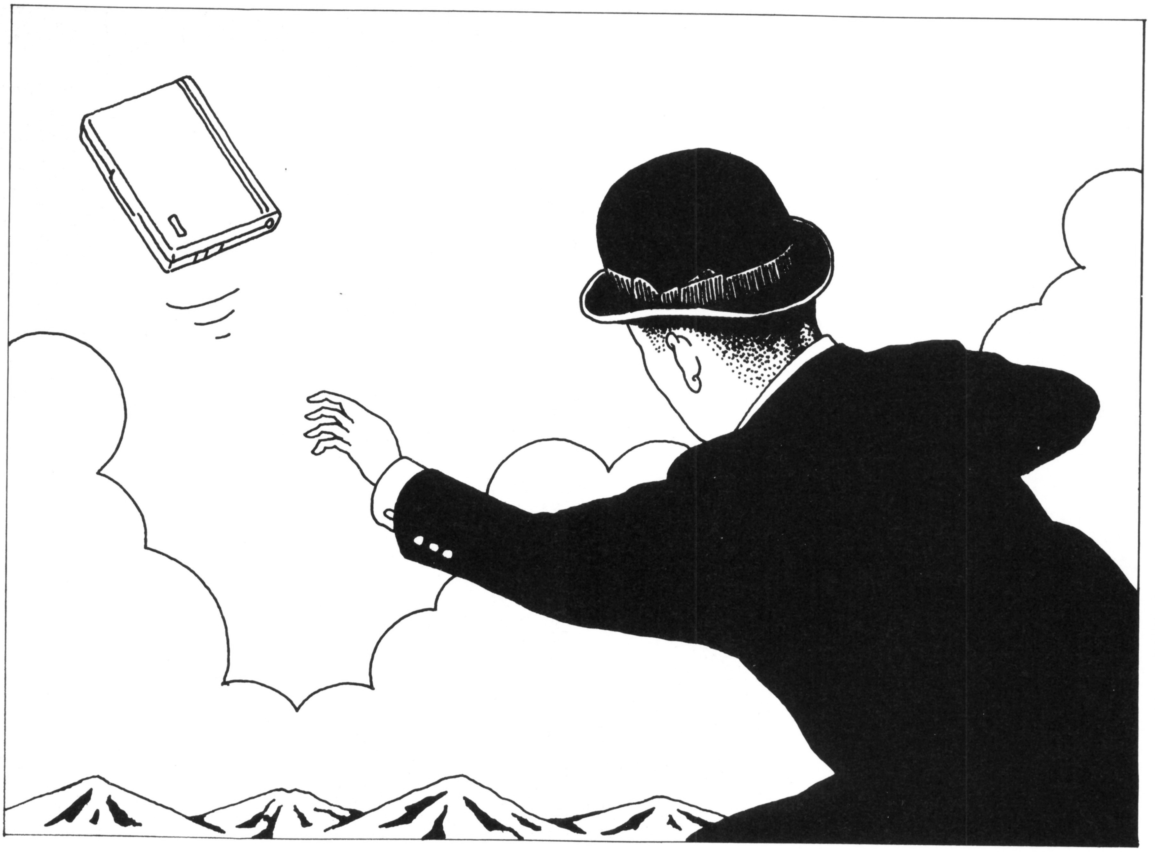
どんな目的に使うにせよ，外に持ち出すのであれば，

重量は最も気になるところだ．ざべ流サブノートの定義では，1.5kgまでという結論が出ているが，現在のところ，その条件に合致するサブノートは数えるほどしかない．また，同じ1.5kgのサブノートでも外部とのファイルのやり取りがあれば，外付けFDDが必須となるし，ACアダプタを持ち歩くことも考えられる．

重量という観点からみれば，1kgしかない日本アイ・ビー・エムのThinkPad 220は他のサブノートから抜きん出て軽い．外付けFDDやACアダプタも軽いため，すべて揃えて1.5kg以下に収まる．これなら十分持ち歩こうという気が起きる．同様に，OmniBook300(あるいはOmniBook425)も外部FDDがないため，携帯時の総重量は1.5kg以内に収めることが可能だ．

逆にもっと割り切ってしまえば，HP100LXという選択もある(HP100LXについては，本誌'93年8月号，9月号掲載の使用レポート(西島範明氏)を参照されたい)．必要最小限の環境を絞り込んだHP100LXは，312gというパソコンとは思えない重量にまとまっている．HP100LXをサブノートと呼ぶのが適切かどうかは分からないが，電子手帳の高機能版という捉え方をすれば，十分実用性は高い．

こうしたサブノートの重量を決定づけているのは，やはりバッテリー駆動時間だ．つまり，バッテリー駆動時間を長くしようとすれば，必然的にバッテリー容量が大きくなり，それに連れて本体も重くなる．そして，サブノートの携帯性が失われてくる．逆に軽くすれば，バッテリー容量を少なくすれば，駆動時間は短くなってしまう．予備のバッテリーやACアダプタを持ち歩く必要が出てきてしまい，結局，総重量は変わらなくなってしまう．すべてのサブノートは，このジレンマに苦しんでるといっても過言ではない．これを解決するのは，ThinkPad 220のようにアルカリ電池駆動のみに限定したり，OmniBook300やHP100LXのようにバックライトを消してしまうという大胆な割り切りが必要になる．重量を取るか，バッテリー駆動時間を取るか，買う側だけでなく，メーカー側も悩むところだろう．

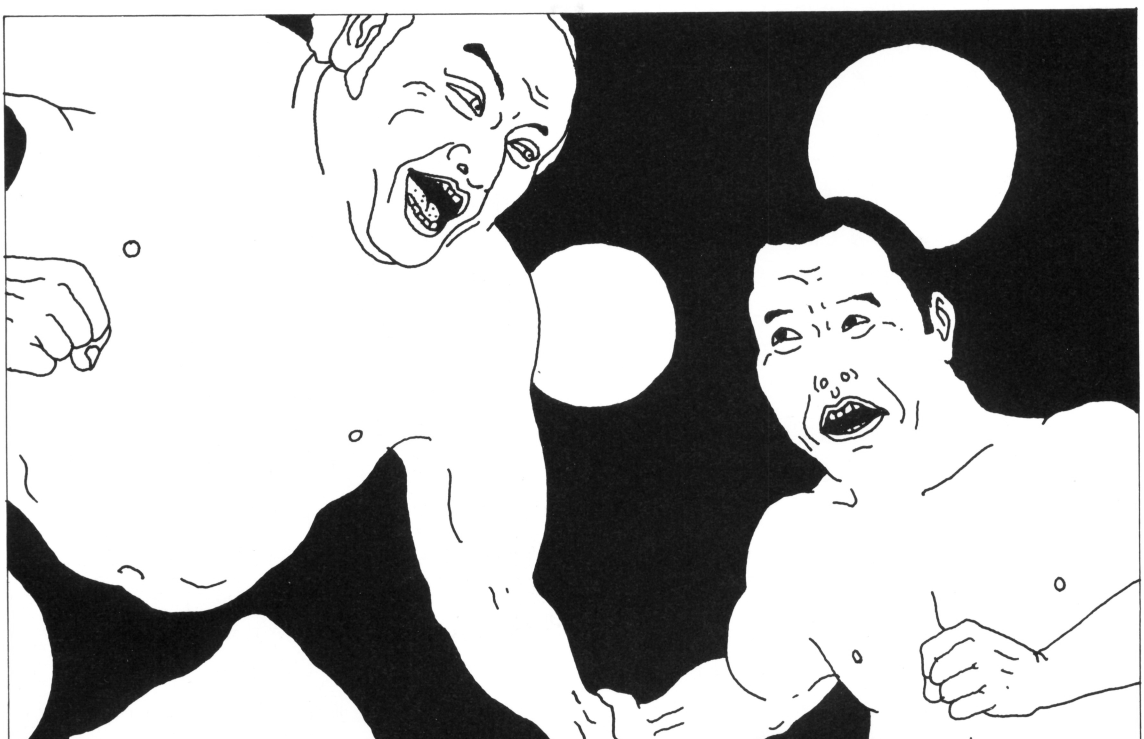


ちなみに，米国でのサブノート事情を見てみると，根本的な体格の違いからか，4ポンド(約1.8kg)を切るかどうかがひとつのポイントになっているようだ．IBMのThinkPad 500が3.8ポンド(1.71kg)，EPSON AmericaのActionNote 4000が3.85ポンド(1.73kg)で，いずれも約400g弱のFDDとACアダプタが提供されており，バッテリーのみで約2時間の連続運用が可能のようだ．カタログスペック上はもう少し長くなっているので，レジュームやサスペンドを使えば，さらに長時間使えるかもしれない．

さて，バッテリー駆動時間についてもう一度考えてみよう．以前，ThinkPad 220のバッテリー駆動時間について本誌で執筆したことがあるが('93年10月号)，やはりThinkPad 220のアルカリ電池のみによる駆動は，かなりストレスがたまる．いつ切れるか，いつ警告音が鳴るかと心配しながら使うのは，精神衛生上よくない．単1や単2の電池パックを自作して駆動時間を伸ばす方法もあるが，結果的にThinkPad 220の軽量・コンパクトというメリットが損なわれてしまう．

やはり，バッテリー駆動時間は長ければ長いに越したことはないのだが，重量の問題を考えれば，ある程度のバランスが必要になる．そのバランスをどのあたりに置いているかで，そのサブノートの方向性が定まってくる．前述のように用途によっても多少の違いは出てくるだろう．例えば，電子手帳的に使うのであれば，何日も連続駆動できる方が望ましいが，外出時にプレゼンテーションをしたり，原稿を数時間書くという目的であれば，そこまで必要はないだろう．

今回の特集⑤で取り上げたマシンは，いずれも2時間近くのバッテリー駆動が可能だ．カタログスペック上では3時間，あるいは4時間と書かれているが，HDDのモーターを回しっぱなしにする条件下ではその数値も半減してしまう．おそらくnenpi.exe*で2時間駆動できるモデルなら，レジュームやサスペンドを併用することで，4時間近くは使えるはずだ．特に，ZEOS ContendaやHANDi-PCのように，メモリ内容をハード



*バッテリー駆動時間を測定する広野忠敏氏作のフリーソフトウェア

ディスクの未使用領域に書き出しておくサスペンド式ならば，実際に運用できる時間はカタログ値以上になることは確実だ．こういった工夫も今後のサブノートに求められる点だ．

その一方で，NiMH(ニッケル水素)バッテリーやリチウム電池など，新しいバッテリーも次々と開発されており，今後はバッテリー駆動時間も長くなり，重量とのジレンマも少しずつ解消されてくるだろう．そのポイントを見きわめるのは，消費者自身である．

CPUパワーとマシンスペック

さて，サブノートの使用目的によって大きく左右されるのは，CPUパワーとマシンスペックだ．CPUについては，Intelのi386SLをはじめ，i486SL，SLエンハンスのi486SX，同i486DXなどが使われている．もちろん，互換CPUも数多く使われており，CyrixのCx486SLC/eやAMDの386SXLVなどを搭載するマシンもある．銘柄は別にして，実際に運用するにはどの程度のCPUパワーが必要なのだろうか．

ざべ流サブノートの定義でも触れたように，まずWindowsなら486，DOSなら386という選択は，ほぼ間違いないだろう．ただ，サブノートの用途から考えて，Windows環境が必要かどうかは疑問である．たしかに，現在のサブノートユーザの多くは，i486DX2/66MHzのデスクトップ機，しかもIBM PC互換機を使っていることが多いといわれている．しかし，それをそのままサブノートに持ち込まなければ，作業はまったくできないのだろうか．

プレゼンテーションなどの用途を考えれば，共通のソフトウェアを使うことができるため，便利だろう．しかし，電話番号を調べるのに，いちいちデータベースソフトを起動するのは，あまり現実的ではない．電話番号をテキストファイル化しておき，DOS上でgrepでも実行した方がはるかに速く快適である．メモを取るにしても，Windows上でノートパッドや秀丸を起動するより，VZエディタなどでパラパラと書く方がいいというユーザも多い．結局，自分がサブノートを使って何をするのかを事前によく把握しておく必要があるということだ．

その他のマシンスペックについても同様のことがいえる．メインメモリやハードディスクの容量は，OSによって決まってくる．まずメモリについては，最大容量もさることながら，どういった形式で増設できるかがひとつのポイントになる．98NOTEなどでは，初期の頃にカード式のものが使われたが，速度的な問題やカードスロットがPCMCIA準拠になった関係から，現在では内部増設タイプのメモリが使われるようになっている．サブノートでもやはり内部増設タイプのものが使われているのだが，絶対的な数が違うため，かなり高価な純正メモリボードを買わなければならない．アイ・オー・データ機器やメルコといった強力なサードパーティがメモリボードを出さないことが大きなデメリットとなっている．しかし，アイ・オー・データ機器やメルコにしても，サブノートは機種によってメモリボードの形状が違う上，市場規模も小さいため，出したくても出せないというのが本音だろう．いっそのこと，サブノートに使うメモリをすべてSIMMモジュール，それもデスクトップ機で使わなくなりつつある30ピンのものにしてしまえばいいのだが，なかなかそういったサブノートは少ない．

ハードディスクについても状況は同じだ．サブノートに使われるハードディスクは，出荷時からマシン内部に固定されているため，保証がなくなるのを覚悟して，本体をバラすしかない．今回取り上げたSWIFTやAV-B5NTのようにパック形式になっていれば，まだ可能性はあるのだが，やはり絶対数からくるコストを考えてもハードディスクの交換は，販売するメリットがなさそうだ．つまり，交換用のハードディスクはまず出てこない，出てきたとしても高価なものになってしまうというのが大方の予想だ．

これらの点を考えれば，メモリもハードディスクも最初からできるだけ多く搭載できた方が望ましいということになる．しかし，実際には大半の機種がメモリが最大16MB程度，ハードディスクは80MBで交換不可能という状態だ．

ただ，こうした拡張性の面で，唯一の望みがあるとすれば，PCMCIAスロットの存在だ．PCMCIAについては，特集③でアホの坂田氏が解説しているので，そちらを参考にしていただきたい．

キーボードとポインティングデバイス

サブノートは，その名前の通り，ノートパソコンよりも小さい．小さいということは，本体に装備するパーツ類も小さくなければならない．例えば，ハードディスクは3.5インチドライブではなく，2.5インチドライブなどが使われている．しかし，これについては，コスト的に高くつくだけの話で，実際の運用にはあまりデメリットはないだろう．

ところが，実際に運用するときに問題になるのがマウスやトラックボールといったポインティングデバイス，そしてキーボードだ．これらの作り方次第では，せっかくのサブノートもまったく使いものにならなくなる可能性もある．

98からDOS/Vへの移行が騒がれて久しいが，そのときに必ず話題になったのがキーボードの配列である．98のキーボードとDOS/Vマシンで使われる日本語106キーボードでは，[Ctrl]キーや[Esc]キーなどの配列が違うため，そう簡単には移れないという話だ．たしかにキーボードが違うと入力しづらくなる．筆者自身もはじめて使ったマシンがEPSON製だったため，いまだにEPSONマシンについているキーボードを愛用している．AT互換機の方も比較的キータッチの近い富士通製日本語106キーボードを使っている．

対象がサブノートとなっても同じことだ．自分が普段使うデスクトップ機のキーボードにできるだけ近いタッチを求めるのはもちろん，ちゃんと両手でキータ



◀キーボードの例として，サブノートの中でもキーが大きいと思われるHANDi-PC（写真上）と，小さい部類に入るQUADERNO 33（写真下）を比較してみた．マシンの外形寸法は，HANDi-PCが255×168.8mm，QUADERNO 33は210×148mm（W×D）である．
写真は2点ともほぼ原寸で，アルファベットキーの大きさはHANDi-PCが13×14mm，QUADERNOは10×10mmとなっている．
HANDi-PCとQUADERNOでは，[Caps Lock]と[Shift]キーの大きさが逆転しているが，このあたりもユーザによっては操作感に影響するかもしれない．
なお，本文中で触れられている[Fn]キーが写真上の左下に見える．

イプできるものが望ましい．ところが，今回取り上げたサブノートを見てみると，キートップのサイズが小さすぎるものや部分的に小さいキーボードなどもあった．これらのキーボードも慣れてしまえば，使えないことはないのかもしれないが，キートップのサイズはデスクトップ機と同等のものが望ましい．

通常，デスクトップ機のキーボードのキートップは，対角線で約1.7cmほどある．これがサブノートになると1.5cm，1.3cmといった具合に小さくなる．わずか数mmの差だが，実際にタイプしてみると操作感がまったく違うことに気づく．

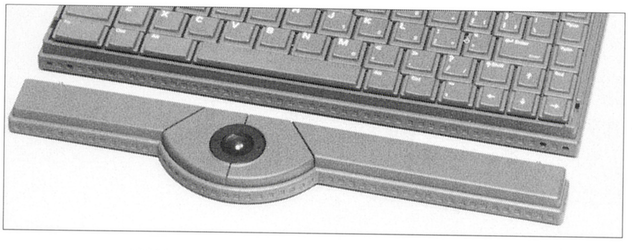
また，キーのストロークやタッチも重要なポイントだ．ノートパソコンの購入ガイドなどでも出てくることだが，本体の大きさ，特に薄さを追及するあまり，キーのストロークが予想外に浅くなっている機種がある．こうしたストロークの浅いキーボードで連続タイプすることはかなり疲れる．

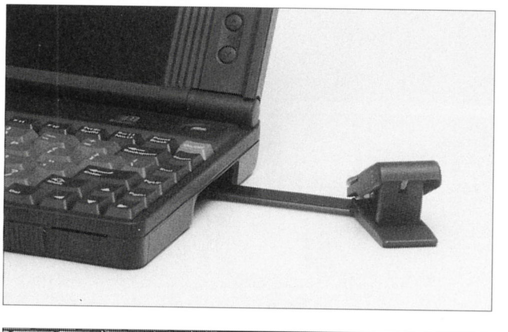
さらに，サブノートではキーの数を減らすため，[Fn]キーを用意し，それと何かのキーを合わせて，別のキーの役割をさせている例もある．たとえば，[Fn]キーと[↑]キーで[Page Up]キー（[ROLL UP]キー）の役割を持たせるといった具合だ．こうした複数のキーによる操作も悪くはないが，できればキートップに何らかの表示がある方が便利だ．

キーボードはユーザとマシンの間に立つ重要なデバイスである．購入時には1行でも文章を入力して，キーボードが自分の手のサイズに合うかどうかを確認するようにしたい．もちろん，大きさだけでなく，キーのタッチやストローク，配列なども同様にチェックするようにしたい．

一方，Windows環境を使う上で，どうしても必要になるポインティングデバイス．ノートパソコン全盛期に比べれば，格段の進歩を遂げ，ほとんどのサブノートがトラックボールを装備するようになった．しかし，トラックボールの操作感も機種によってまちまちで，片手で操作できるものもあれば，両手でも操作しづらいものもある．特に，キーボード上部ではなく，本体の側面にトラックボールのボタンを配置したマシンは，ボタンの強度やホコリからの保護などの点で不安が残る．また，ボールの大きさや材質についても工夫が足りない．AppleのPowerBookシリーズでボールをベアリングの玉に交換するというアイデアがあるそうだが，サブノートでも同様の工夫をしないと，動きが軽すぎて使いにくいのだ．

トラックボール以外のポインティングデバイスとしては，IBMがノートパソコンで採用しているTrack-Pointが注目できる．キーボードの[G]と[H]と[B]の間にある赤い小さなスティックでポイントを動かし，スペー









▶ポインティングデバイスについては，各メーカーとも工夫しているようだ．ほとんどのマシンがトラックボールを採用しているが，マシンによって，配置されている場所が違う．使用感は，ボールの位置だけでなく，ボールとボタンの位置の相関によってもずいぶん変わってくる（写真上左よりZ-LITE，OmniBook 300，下左よりTWINHEAD，SWIFT）

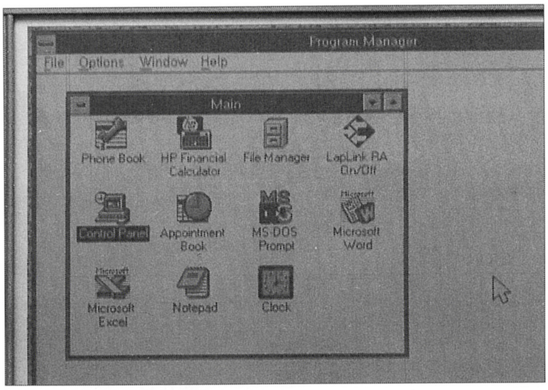
スバーの手前にあるボタンをクリックする．これをぜひともサブノートでも採用してほしいところだ．また，OmniBook300で採用しているポップアップマウスも魅力的なデバイスだ．いずれにしても，現状のトラックボールは最低限のものでしかなく，メーカー側もあまり重要視していないのが実状だ．キーボード並み，いやWindows環境を考えれば，キーボード以上に工夫をこらしてほしいところだ．

日本語環境とOSはどうする?

さて，国内で販売されているサブノートは，EPSONのPC-486Portableを除けば，すべてIBM PC互換機だ．ということは，日本語を表示するためには，おなじみのDOS/Vが必要となる．そのため，ほとんどのサブノートがマイクロソフト版MS-DOS/Vを付属している．実際の運用上はあまり問題ないかもしれないが，フリーソフトウェアなどを利用することを考えたとき，IBM DOSバージョンJ5.0/V（IBM版DOS/V）を前提にしているものが多いため，一抹の不安は残る．これはWindows関しても同様のことがいえそうだ．もし，可能であれば，IBM版DOS/Vとマイクロソフト版MS-DOS/Vのどちらかを選べるようになっているとありがたい．

もうひとつの問題として挙げられるのが，OmniBook 300やQUADERNO 33のような例だ．OmniBook300はご存じのようにROMで英語版のDOS5.0やWindows，Excel，Wordなどを持っている．QUADERNO 33も英語版DOS5.0がROMで供給されている．DOSであれば，フリーソフトのディスプレイドライバを組み込むことで日本語表示は可能になるが，誰もが簡単にできるわけではない．そのため，QUADERNO 33では日本オリベッティが日本語化するためのDocumentとファイルを添付している．

しかし，Windowsの場合はこうした対処ができないため，海外のサブノートを購入する足かせにもなっている．そこで期待されるのが，C.F.Computingが開発中の『Win/V』だ．Win/Vは英語版Windows上で日本語Windows用のアプリケーションソフトを動作させるためのもので，OmniBook300などでも動作可能といわれている．現在，βテストが進行中だが，おそらく本誌



▲OmniBook300はDOS，Windowsに加え，Excel，WordなどをROMでもっている．こうした形態がサブノートPCの指標のひとつあることは間違いない．が，これらはすべて英語版なのである．ユーザにとっては，Win/Vの一日も早い登場が待たれる

「DOS/V Paradise」などでもC.F.Computing代表の西川氏自身から何らかのアナウンスがあるだろう．正式発売を楽しみにしたいところだ．

さらに，日本語化されたソフトウェアもサブノートをあまり意識していないものが多いのも残念な点だ．ワープロや表計算などの定番ソフトは充実しているのだが，サブノートで使いたいPIM（Personal Information Manager）ソフトがほとんどない．あったとしても非常に使いにくく，現実的な面を考えると，Worksなどの統合ソフトなどに頼るしかない．本誌’93年12月号でZEOS Contendaにバンドルされてきた“Lotus Organizer”を紹介したが，ロータスにはぜひこれを日本語化していただきたい．また，他のソフトハウスにはこのLotus Organizerに負けない優秀なPIMソフトを開発して欲しい．PIMソフトの充実なくして，サブノートの普及・進歩はないといっても過言ではないだろう．

●

サブノートは，’93年になって突如として注目を浴びたジャンルである．潜在的なユーザは多いのだが，正直なところ，二の足を踏んでいるユーザも少なくない．商品自体も成熟度が低いものがいくつかあり，改良する余地も多数残されている．ぜひともメーカーにはユーザの不満を聞き入れ，よりよい商品を開発していただきたい．もちろん，それに付随するソフトウェアの開発も急務だ．

各メーカーの新製品なども含め，今後，どういった展開を見せるか，じっくりと注目をしたい．

特集・サブノートの気になるあのこと

3

PCMCIAの周辺

●著者●
アホの坂田

●概要●
日本でも最近になってPCMCIA関連の話題がパソコン雑誌などに登場するようになった．
しかし，海外での普及ぶりを考えると，まだまだこれから，というのが現状だ．
本稿では，PCMCIA全般の解説を行うとともに，
問題点，将来の展望などについても触れてみたい．



日本ではまだPCMCIAスロットを持つPC自体が少ない．よく知られるところでは，IBMのThinkPadシリーズや，オタクの間でわけの分からない(?)売れ方をしているHP100LX，7月に発表されたPC-9821Neなどといったところだろうか．むろん，この中に本特集でも紹介されている各社サブノートPCが含まれるわけだが，日本の現状を考えた場合，NEC，EPSONなどのサブノートが普及するようになって，初めて多くの日本人もPCMCIAカードを使うようになり，海外で頻発しているトラブルにも遭遇するようになるだろう．この国はなんでも3年は余計にかかるのだ．＼(^o^)／

PCMCIA設立と趣旨

いまさら説明するまでもなく，PCMCIA(Personal Computer Memory Card Intenational Association)はITU-T(国際電気通信連合-電気通信標準化部門．旧CCITT)やISO(国際標準化機構)などのような国際機関ではなく，任意の民間団体である．1989年の発足以来，加入メンバーは増え続け，現在では約500にも上る多くの企業が参加している．日本にもJEIDA(Japan Electronic Industry Development：日本電子工業振興協会)という団体があるが，雑誌などでよく“PCMCIA/JEIDA”と書かれているのは，PCMCIA側とJEIDA側双方でできるだけ規格を統一しておきたいという意思の表れである．

賢明な読者ならお気づきになると思うが，PCMCIAという言葉そのものを日本語訳すると「パーソナルコンピュータメモリカード国際なんたら」ということになって，どうしてこんなものが世界のモデムやLANカードに大きな影響を与えているのかと不思議に思うであろう(話の都合上，勝手に振っている)．実はここが，アメリカを筆頭とする外国勢のずるさである．

もともとPCMCIA1.0に相当するJEIDA4.0は日本で規格と製品が出たものであり，三菱や富士通，東芝など，メモリカードでは日本が先頭を切っていたのである．で，先を越された格好の外国勢としては「ふむふむ，なーんだ，これって単にBUSに接続したメモリなだけじゃん，なら俺たちゃ，これにI/O機能も入れてISAにもEISAにも使える周辺機器も作っちゃうもんね」とい

うことで，規格をパクッって作ったのが，PCMCIA2.0である(ほんまかいな).

PCMCIAができる前はPOCKET COMPUTER社と提携した富士通，メモリカードを作っていた三菱，東芝など，多くの日本企業が活躍したが，その後の主導権は完全にアメリカが握っている．バスからしてPC-9800などは世界中の誰からも相手にされてないからこうなって当たり前なんだけどね．(^^;

サニーベールにあるPCMCIA本部の公式資料では，以下のように自身を説明している．

「PCMCIAは1989年に集積回路(IC)の標準化を確立し，耐久性，小電源消費，軽量化が不可欠である様々なコンピュータと，他の電子機器間の互換性を促進するために創設されました．現在，PCMCIAはポータブルやパーソナルコンピュータに限らず，例えばデジタル画像(カメラ)や情報販売機をも含む，すべてのコンピュータ間の互換性を目指しています．PCMCIAの会員は広範なコンピュータ産業のセグメントを代表し，ハードウェア，ソフトウェア，半導体部品，コネクタ，コンピュータ周辺機器およびその関連する産業を含んでいます．

PCMCIAは急速に成長し，市場駆動された標準化団体であり，積極的にPCカード標準化の国際市場での受け入れと採用を促進するとともに，市場の迅速な開発を目指しています．当協会はPCカードの国際標準化を確率，維持し，教育を通して標準化を促進するという二つの主要な短期目標を掲げています．

これら目標達成のため，PCMCIAはメモリモジュールとパッケージの標準化を目的とするJEIDA，EIA(Electronic Industries Association：米国電子工業協会)，JEDEC(Joint Electronic Device Engineering Council)委員会や，ISOなどの他の産業組織や標準化グループと密接な連絡を取り合っています」(日本語ガイドより抜粋)

というのが設立趣旨であるが，太平洋戦争といい，株の大暴騰，大暴落といい，外人はいつも巧妙な後注釈をつけるのがうまいなぁと感心させられる．

なお，PCMCIAについての最新情報は，以下に連絡すれば300ドル程度で入手できる．

PCMCIA
1030G East Duane Avenue Sunnyvale,CA 94086 U.S.A.
TEL:408-720-0107 FAX:408-720-9416

PCMCIAの目的

PCMCIAはその設立趣旨でも述べているように，これから巨大な市場に成長するであろう携帯用PCと移動体通信を最大のターゲットとしている．コンピュータと通信はやがて世界中の仕事やライフスタイルを確実に変えてゆくだろうが，そのためにはコンピュータをド素人でも簡単に使えるようにする必要があるのだ．

IRQやインタラプトなどを苦もなく設定する人間などはご遠慮願うのだ．＼(^o^)／　このため，PCMCIAはいろいろな工夫をやっているが，ここ数年の目標は完全な“Plug & Play”の実現だろう．Plug & PlayはPCの機能の拡張をPCMCIAカードで自由にやれるというものだ．たとえPCの電源が入っている状態でも自由にカードを抜き差しすることができ，さらに挿入されたカード自身の種類と目的をPCのアプリケーションが自動で認識し，アプリケーションが自ら最適の状態で起動することまでやるのである．こうなって初めてPCはステレオやCDプレーヤなどのように，一般の人間まで購買層を広げることができ，かつ，どこでもいつでも使えるようになるだろう．

そのためにPCMCIAはPC側とカード側にいろいろな工夫をしている．では，実際にそれらを見てみよう．

PCMCIA1.0と2.0

PCMCIAの話をするときによく間違えられているのが規格のバージョンと，形状・種類の数字である．日本の雑誌では，ひどいときには通常のSRAMまでPCMCIA2.0などと写真付きで雑誌などに紹介されていたりする．(^^;

PCMCIA1.0/2.0というのはPCMCIAの会議で決められた仕様規格そのものの呼び名でであって，形状などの名称ではない．PCMCIA1.0はPCMCIAが発足後まもなく，メモリカードの仕様を中心に決められたものである．

その後，この仕様をもとに従来のISA，EISAなどのI/O機能をもつ周辺機器ボードをカード化するために決められたのがPCMCIA2.0なのだ．

現在，PCMCIAは2.01まできているという話だが，ひょっとしたらIntelのExCAなどの仕様が入ったものになったのだろうか．

いずれにしてもようやく真のPlug & Playを目指す動きとなっている．

PCMCIAカードの種類と規格

PCMCIAカードは既存のISA，EISA，MCA，VESA(VL-Bus)，PCIバスと互換性を持っており，OSとデバイスから独立している．形状的には以下のサイズが規定されている．

●PCMCIA TYPEⅠ，TYPEⅡ，TYPEⅢ

PCMCIAでは現在，形状として**図1**のような三つのタイプを規定しており，将来的にはTYPEⅣまで検討されている(JEIDA4.2では16mm厚のTYPEⅣが発表されている)．

まず，TYPEⅠであるが，これは外形54×85mm，厚さ3.3mmのカードで，I/O機能を持たないメモリカード(例えば初期のHP95LXのメモリ)などに使われている．

ピンの数は68ピンという仕様が決められ，これが標準となった．

TYPEⅡはTYPEⅠのアッパーコンパチの製品であり，カードの厚さは5mmとなっている．ただし，68ピンのコネクタ部分の形状はTYPEⅠと同様であるので，3.3mmである．

TYPEⅡになって初めてフルスペックのI/O機能を搭載するようになり，これによりFAXモデムやイーサネットカードなど，従来のISAバス用拡張ボードがPCMCIAカードの形で供給されるようになった．現在，最も多く普及しているPCMCIAカードはTYPEⅡのものである．

そして1992年秋になってPCMCIAはハードディスクのような駆動系デバイスまでもカードにする規格を決めた．これがTYPEⅢと呼ばれるもので，TYPEⅡの5mm厚を10.5mmまでに拡張したものとなっている．

しかし，TYPEⅠ，TYPEⅡ，TYPEⅢも違うのはカード厚で，どのカードも先端のエッジ部分(3.3mm)は標準のコネクタに接続できるようになっている．

●ピン配列とPC本体との接続

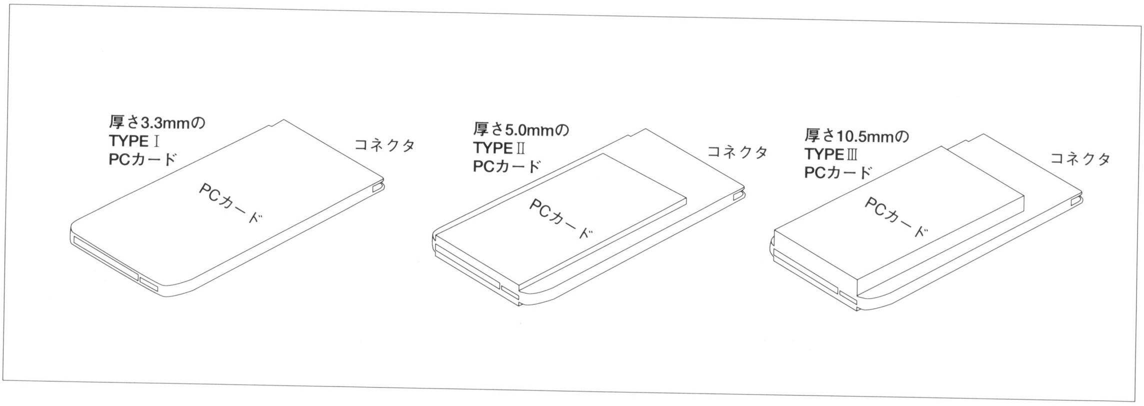
PCMCIAカードのコネクタ部分は34ピンが並列に並んだ形状になっているが，ピンの中心と隣のピンの距離は上下左右とも1.27mmと規定されている．PC側のコネクタはオス，カード側はメスとなっている．カード側のコネクタの深さは3.2mmであるが，例外として電源とアースのコネクタのみが3.6mmと深くなっている．つまりすべての信号線がちゃんと接続されてから電源が入るしくみとなっているのだ．

さらにPC本体がカードの挿入を認識できるようにするため，36番ピンと67番ピンは2.6mmと，普通の信号線より浅くなっている．

このような仕組みにより，PCMCIAカードはPC本体の電源が入っている最中に抜き差ししてもPCとカードを守る安全な設計となっているのだ．

ここでは代表的なメモリカードでのピン配列を以下

図1●PCMCIAカードの形状



＊PCMCIA規格の対象をAT互換機に限定することによって，互換性を高めることを目的した規格

に示し，次に現在，PCMCIA2.0仕様のフラッシュメモリカードとして全世界で圧倒的なシェアをもっているSunDiskの構造などを覗いてみる(**表1**参照)．

SunDiskのフラッシュメモリカード

最近，本誌や「PC WAVE」「アスキー」，はたまたNIFTY-ServeのFMODEMなどで大騒ぎになっているHP100LXであるが，このメモリカードの大半はSunDisk製である．日本ではエプソン販売が総代理店となっており，筆者が100LXで書いているこの原稿も，実はSunDiskのフラッシュメモリカード(FLASH-PACKERという名称で販売されている．**写真1**)．SunDiskのフラッシュメモリシリーズはすべてPCMCIAの規格に基づいて設計され，現在，最大20.9Mバイトの容量※を持つTYPE IIのものが発売されている．※

これらのカードにはPC本体と通信するためのSunDiskオリジナルインテリジェントコントローラが搭載されている．このインターフェースを通じて，PCはカード上の512バイト単位のメモリブロックにアクセスする(**図2**)．したがって，PC本体はSundiskを通常のIDEハードディスクとして認識する格好になっている．

SunDiskのセクタサイズは512バイトである．つまり，PC本体は通常のIDEだと思ってRead/Writeすればよく，特別な工夫は何もいらない．加えてSunDiskには通常のハードディスク以上のデータセキュリティ機能が組み込まれていることは特筆に値する．例えば通常のハードディスクではパフォーマンスの低下を恐れるため，データの読み込み時にいちいちRead/Writeしてベリフ

写真1●FLASH-PACKER SDP5-20



表1●一般的なメモリカードのピン配列

PIN	SIGNAL	I/O	FUNCTION	+/-
1	GND	Ground		
2	D3	I/O	Data bit 3	
3	D4	I/O	Data bit 4	
4	D5	I/O	Data bit 5	
5	D6	I/O	Data bit 6	
6	D7	I/O	Data bit 7	
7	CE1	I	Card Enable	-
8	A10	I	Address bit 10	
9	OE	I	Output Enable	-
10	A11	I	Address bit 11	
11	A9	I	Address bit 9	
12	A8	I	Address bit 8	
13	A13	I	Address bit 13	
14	A14	I	Address bit 14	
15	WE/PGM	I	Write Enable	-
16	RDY/BSY	O	Ready/Busy	+/-
17	Vcc			
18	Vpp1		Programming Supply Voltage 1	
19	A16	I	Address bit 16	
20	A15	I	Address bit 15	
21	A12	I	Address bit 12	
22	A7	I	Address bit 7	
23	A6	I	Address bit 6	
24	A5	I	Address bit 5	
25	A4	I	Address bit 4	
26	A3	I	Address bit 3	
27	A2	I	Address bit 2	
28	A1	I	Address bit 1	
29	A0	I	Address bit 0	
30	D0	I/O	Data bit 0	
31	D1	I/O	Data bit 1	
32	D2	I/O	Data bit 2	
33	WP	O	Write protect	+
34	GND		Ground	
35	GND		Ground	
36	CD1	O	Card detect	-
37	D11	I/O	Data bit 11	
38	D12	I/O	Data bit 12	
39	D13	I/O	Data bit 13	
40	D14	I/O	Data bit 14	
41	D15	I/O	Data bit 15	
42	CE2	I	Card Enable	-
43	RFSH	I	Refresh	
44	RFU		Reserved	
45	RFU		Reserved	
46	A17	I	Address bit 17	
47	A18	I	Address bit 18	
48	A19	I	Address bit 19	
49	A20	I	Address bit 20	
50	A21	I	Address bit 21	
51	Vcc			
52	Vpp2		Programmig Supply Voltage 2	
53	A22	I	Address bit 22	
54	A23	I	Address bit 23	
55	A24	I	Address bit 24	
56	A25	I	Address bit 25	
57	RFU		Reserved	
58	RESET	I	Card Reset	+
59	WAIT	O	Extend bus cycle	-
60	RFU		Reserved	
61	REG	I	Register select	-
62	BVD2	O	Battery voltage detect 2	
63	BVD1	O	Battery voltage detect 1	
64	D8	I/O	Data bit 8	
65	D9	I/O	Data bit 9	
66	D10	I/O	Data bit 10	
67	CD2	O	Card detect	-

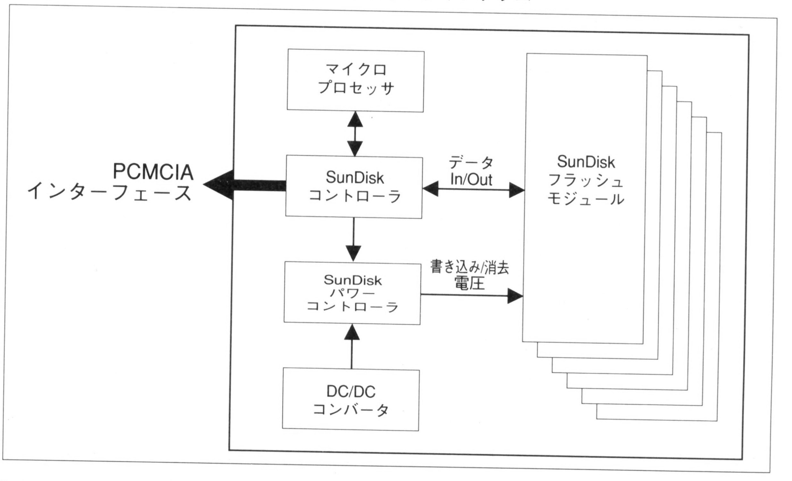
※20MバイトのSDP5-20の価格は173,500円．なお，従来SunDiskのフラッシュメモリシリーズには駆動電圧の違いなどにより，SDP，SDPL5，SDP5の3シリーズがあったが，今後はSDP5シリーズに一本化される．

ァイは行わないが，SunDisk SDP5シリーズはこれを行っているのである．

そして万が一，ビット不良個所が発見されるやいなや，これを避け，スペアビットを代替として使うのだ．同時にそれぞれのセクタが何回利用されたかもカウントし，メモリセルを平均化して使うような仕掛けまでやっている．

驚くべきことに不良セクタが発生した場合，そこを使用禁止とし，自動的にスペアセクタにデータを移すようなことまでやっているのである．SunDiskの中は極めて精巧なオートメーション管理システムなのだ．

図2●FLASH-PACKER SDP5シリーズのブロックダイアグラム



このため，SunDiskのデータ信頼性は非常に高く，筆者などはこれを聞いて以降，安心してバックアップも取っていない．(^^;(あ，危ない)

SunDiskのセクタへの書き込み耐用回数は20万回(読み込み回数は無限)とされているが，この回数は1年365日毎日8時間，HP100LXなどで使った場合，20年間は保証できる回数だという．

SunDiskの工夫はこれらデータ管理だけに留まらず，消費電力にまで及んでいる．一般にノートパソコンやポケットコンピュータ，PDA(Personal Digital Assistant)などで最も要求されるのは，動作スペックよりも重量や電池寿命であるが，とりわけ電池寿命は非常に重要である．筆者がなぜキーボードの打ちにくいHP100LXを使っているかという最大の理由は，他のPCでは真似のできない超ロングな電池持続時間にある．

表2●FLASH PACKER SDP5シリーズのシステムパフォーマンス

Model		SPD-XX
Start Up Tomes	Sleep to write: Sleep to read: Reset to ready:	<2.5ms maximum <2.0 ms maximum 50msec typical 400msec maximum
Delay Active to Sleep		Programmable
Data Transfer Rate To/from Flaxsh		3MBytes/sec burst
Data Transfer Rate To/from Host		60MBytes/sec burst
Cntroller Overhead	Command to DRQ	<1.25ms maximum

なにしろ毎日使っても1カ月は楽勝なのである．持続時間のケタが違うのだ．これはSunDiskを入れて使ってもあまり変わらないので不思議に思っていたのだが，やはりSunDisk側に工夫がしてあって，**表2**のように数ミリ秒単位でカード自体をスリープモードにする仕掛けを実現しているのだった．

PCMCIA2.0のフラッシュメモリカードの例としてSunDiskを取り上げたが，こんな小さなメモリカードにも膨大な工夫がされていることが理解できる．

PCMCIAカードのID情報格納庫

PCMCIAの目指すところがPlug & Playであるならば，ユーザはこれまで設定や機能拡張などが非常に難しかったPCを，まるでそのへんのカメラやラジカセと同じような操作で使えるということになる．

アプリケーションの起動さえもPCMCIAカードが自動でやってくれるようになるのだ．このように，これまでのPCでは考えられなかったことをやるわけであるから，裏方として動くPCMCIAソフトウェアにかなりの工夫が必要なのはいうまでもない．人間が楽をする分，コンピュータは常に大変なのだ．

PCMCIAのソフトウェアというと，主にPC側の認識ソフトウェアをいうが，その前に「自分がいったい何者であるか」をシステム側に知らせるためのPCMCIAカード側の素性を記録する部分に触れておかねばならない．

●CIS(Card Identification Structure)とタプル(Tuple)

PCMCIA2.0になっていろいろな用途のPCMCIAカードが開発される事を想定してカードの一部に素性を記録するメモリ空間を作ることになった．これが，CIS(Card Identification Structure)もしくはmetaformatと呼ばれる部分である．SISはスタック構造を形成し，それぞれのスタックはタプルという128バイト単位のデータブロックで構成されている．

タプルには**表3**のようにデバイス情報やコンフィグレーションレジスタの番地，IRQや割り込み情報などが記録される．つまりPCは起動もしくはPCMCIAカードが挿入された時点で最初にこれらの値を参照して，カードがどんな用途のカードなのか，何を要求しているか，などを判断するわけである．

では，それらの情報を判断する側のPC側のソフトウェアを見てみよう．

PC側のソフトウェア

PCMCIAカードを自動認識し，それらの要求するリソースを満たし，IRQや割り込みを自動で用意するなどということは，古くからのIBM PCユーザにとっては，ほとんど「冗談でしょう」というくらい難しいものである．

なぜならば，それらはBIOSやデバイスドライバが管理するもので，PC自体が起動している状態からでは再設定不可能なものだったからである．

そのため，PCMCIAではBIOSやデバイスドライバを従来との互換性を図りながら新しく作り直すような作業に着手したわけであり，現在も検討が続けられている．

Intelと某社(^^;の2社のソフトウェアブロックダイアグラム図を参照してみよう(**図3**)．

表3●CIS内のタプル情報例

CODE	NAME	DESCRIPTION
0	CISTPL_NULL	Null tuple; ignore
1	CISTPL_DEVICE	The device-information tuple(Common Memory)
2-7	CISTPL_CHECKSUM	(Reserved for future, upward-compatible versions of the device-information tuple)
8-0Fh	CISTPL_LONGLINK_A	(Reserved for future,incompatible version of the device-information tuple.)
10h	CISTPL_LONGLINK_C	The checksum-control tuple
11h	CISTPL_LINKTARGET	The long-link-control tuple(to Attribute Memory)
12h	CISTPL_NO_LINK	The long-link-control tuple(to Common Memory)
13h	CISTPL_VERS_1	The link-target-control tuple
14h	CISTPL_ALTSTR	The no-link-control tuple
15h	CISTPL_DEVICE_A	The Level-1 version/product-information tuple
16h	CISTPL_JEDEC_C	The alternate-language-string tuple
17h	CISTPL_JEDEC_A	Attribute Memory device information
18h	CISTPL_CFIG	JEDEC programming information for Common Memory
19h	CISTPL_ENTRY	JEDEC programming information for Attribute Memory
1Ah	CISTPL_DEVICE_OC	The configurable-card tuple
1Bh	CISTPL_DEVICE_OA	The configuration-entry tuple
1Ch	CISTPL_VERS_2	Other operating conditions device information for Common Memory
1Dh	CISTPL_FORMAT	Other operating conditions device information for Attribute Memory
1Eh-3Fh	CISTPL_GEOMETRY	(Reserved for future standardization)
40h	CISTPL_BYTEORDER	The Level-2 version tuple
41h	CISTPL_DATE	The format tuple

図3を見ると，両社の間に微妙な違いが見られるのが分かるだろう．すなわち，PCMCIAは現在進行形で仕様を拡張しており，それでも発生するさまざまなトラブルをPCメーカーが必死でサポートしている，というような状況なのだ．

その結果，PCMCIAも知らないうちに勝手に生まれて各所で出まわっているのが“Enabler”と呼ばれるドライバの1種であり，通常のソケットサービスやカードサービスでは認識できない場合に，PCMCIAカードメーカーなどがパソコン通信などを通じて配布していたりする．Enablerについては「PC WAVE」'93年10月号に山路トトロという人が非常に詳しく解説しており，PCMCIAの資料くらいしかもっていない筆者は驚くばかりであった．きっと，PCMCIAで苦労されている開発者なのだろう．早期にEnablerの存在がPCMCIAから消えるときが来るのを望むばかりである．

さて，話をもとに戻すが，PC側のPCMCIAのソフトウェアはソケットサービス(Socket Service)とカードサービス(Card Service)という二つに大別できる．PC上のアプリケーションやデバイスドライバなどはこの二つのAPIを通じて従来の周辺機器ボードと同じようにPCMCIAカードにアクセスできるわけである．

●ソケットサービス

ソケットサービスはいわばPCのBIOSの拡張モジュールともいえるもので，PCはソケットサービスを通じて

カードがシステムに挿入されたのか，あるいは抜かれたのかを認識する．すでにAMI，PhoenixなどのBIOSメーカーは自社BIOSにソケットサービスの機能を搭載したものを出荷しているが，多くはまだデバイスドライバもしくはAUTOEXEC.BATで実行し，常駐する形式で供給されている．実際のソケットサービスの実行はソフトウェア割り込みINT 1AHを用いる．

驚くべきことにPCMCIAの仕様では1台のPCには16枚ものPCカードを接続することが可能であり，通常のPCと同様，IDEコントローラカードには2台のハードディスクを接続することが可能，SCSIアダプタに至っては7台までのデバイスを接続できるという．

このため今後はOmniBook300が持つ以上の数(4スロット)のPCMCIAカードスロットを持ち，あらゆる用途に対応した究極の携帯用PCも出現する可能性がある．

●カードサービス

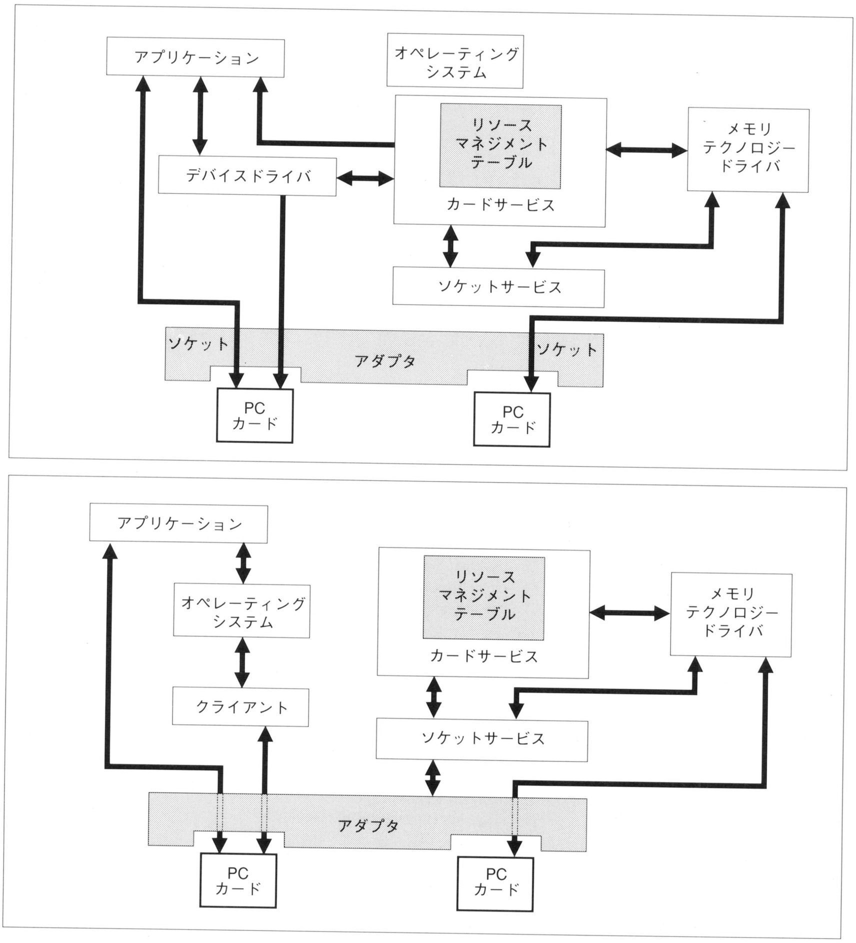
ソケットサービスによりPCMCIAカードの抜き差し状況を認識したPC側は，次にカードサービスドライバを通じて実際にPCMCIAカードにアクセスする(この場合のソフトウェア割り込みもソケットサービス同様，INT 1AHであるが，当然ながらカードサービスの方がファンクションの種類は多い)．

カードにアクセスしたPCは前出のCIS内のタプル部分を参照し，カードがどんな用途のカードなのか，どんなリソースを要求しているか，I/Oポート，IRQはどこを使いたいのか，などを判断するわけである．

しかし，ひとくちに言って，この部分は非常に高度かつ困難を伴う仕事である．なぜならば，PCで用意されているリソースやI/Oポート，IRQなどは無限ではなく，実に限られたものであり，これをPCMCIAカードの抜き差しも感知しながら，変幻自在に管理するのであるから．もし，どこか1カ所でも割り込みを間違えれば，PCはたちまちハングアップするだろう．

PCMCIAではこれらの問題に対処するために，カードサービスドライバ内のモジュール内部に“Resource Management Table”というリソース管理テーブルを用意し，システム起動時からPCのすべてのリソース管理を行うような仕様にしている．これらの管理は本来がBIOSのすべき仕事であるので，将来的にはOSそのものに組み込まれるだろう(すでにOS/2やWindowsNTはそうなっているいるという話もある)．

図3●Intel(上)と某社(下)のブロックダイアグラム



このリソース管理テーブルによって，PCもしくはアプリケーションソフトとPCMCIAカードは常に情報を交換して，ソケットサービスとともに円滑なシステム運営を目指すわけである．

ちなみに筆者はHP100LXでこの原稿を書いているが，100LXのソケットサービスとカードサービスは極めて優秀で，非常に多くのPCMCIAカードを認識する．その100LXの起動時には以下のようなメッセージが表示されるが，これらのメッセージを見るとソケットサービス，カードサービスが実際にシステム起動時に100LXにインプリメントされる様子がよく分かる(**図5**)．

100LXは本体サイズがギネスブックに登録されるほど極小だけに，ソケットサービスとカードサービスが動くさまは実に ケナゲである．＼(^o^)／

PCMCIAの問題点

以上，見てきたようにPCMCIAはこれから始まるPCの大衆化と携帯性という分野で最も重要な役割を果たすものと期待されている．ただ，現在はPCのアーキテクチャの見直しやOS戦争，CPU戦争など，PCそのものの存在が極めて流動的であるので，PCMCIAもその影響を大きく受けている．このため，せっかく購入したPCMCIA製品が，PCのメーカーや機種によって動作しないというトラブルが頻発しているのが実態である．

米国のある雑誌のレポートによると，数十種類のPCと数十種類のPCMCIA製品を組み合わせ，ちゃんと動作するかどうかをテストしたところ，なんと半分以上の組み合わせで動作しなかったという驚くべき結果も発表されている．恐らくPC側は各PCMCIAカードメーカーのEnablerで収拾もつかない状況になったのではなかろうか．(^^;

PC/ATアーキテクチャ一本の米国市場でもこんな状態なので，PC-9800，EPSON，DOS/Vマシンなど異なるアーキテクチャのマシンが多数存在する日本では，米国市場以上の大混乱が発生する可能性もある．

PCMCIA製品の種類

PCMCIA製品はすでに米国市場でかなりの種類と数

図5●HP100LX起動時に現れるメッセージ

```
Microsoft Device Driver for BIOS Service Version 2.0
The Device Driver system installed 02 drive(s) a: - b:

HP Card Instration Cliant Version 1.08
```

の製品がリリースされている．参考資料として米国市場で発売中のPCMCIAスロットを装備したマシンとPCMCIAカード製品をピックアップしてみたので参照してほしい(**付録1**)．

製品の種類としてはメモリカード，FAXMODEMカードなどを筆頭にLANカードやハードディスク，SCSIカードなど，従来のボードで出ていた製品のほとんどがPCMCIAカードの形で製品化されている．これらは1993年8月頃までのものなので，本号が発売されるころにはさらに多くの製品が発売されているはずである．

PCMCIAの将来

PCMCIAの最終的なゴールは前に述べたように自由でフレキシブルなPlug & Playの実現である．PCの機種を問わず，CPUとOSを問わず，さらにアプリケーションを問わず，Plug & Playを実現するのは世界で勃発している民族紛争の平和的解決と同レベルの難易度に近いものがあるだろう．(^^; PCMCIAは現在も拡張し続け，PCが進化を続ける限り，どこまでいってもPCMCIAは「PCMCIA準拠」にならざるを得ないと思われる．

しかし，PCがこの世に出てすでに15年以上が経過し，マニアや技術者だけの道具からやっと一般の人の仕事や生活の中に定着を開始したことを思うと，これからの数年の新ユーザ層の開拓，PCの使いやすさの追求においてPCMCIAの果たす役割は非常に重要であろう．

〈参考資料〉

・特集「サブノートは街に出る」，『PCWAVE』'93年10月号，電波実験社
・“PC MAGAZINE”，Ziff-Davis
・“SDP5 SERIES PRELIMINARY MANUAL”
・“PCMCIA Glossary”，

付録1●PCMCIAカード関連製品リスト（※Data Sources ReportCOPYRIGHT ZIff-Davis Publishing Co. 1993）

付録1-1●PCMCIAスロットを持ったPC

製品名	会社名	電話番号	FAX番号
BICOM B260i Notebook	ABC Computer（USA）Corp.（subsidiary of ABC Computer Co., Ltd.）	800-336-2280; 310-325-4005	310-325-6369
LC8600	Abstract R&D	510-253-9588	510-253-9511
AcerAnyWare V386SL	Acer America Corp.（subsidiaty of The Acer Group）	800-SEE-ACER; 408-432-6200	408-456-0471
Altima Traveler	Altima Systems, Inc.	800-356-9990; 510-356-5600	510-356-2408
Alt-sx25	Altura Computer Systems	719-685-5502	719-685-5356
MiTAC 1600A	American Mitac Corp.（subsidiary of MiTAC International Corp.）	800-777-1688; 510-623-5300	510-623-5390
APF4000 APF4300	APF	803-244-4416	803-879-2030
AMS TravelPro 386-25SL AMS TravelPro 486-33 Color AMS TravelPro 486-50 Color（※） AMS TravelPro 486-50 Monochrome AMS TravelPro 486-66 Color AMS TravelPro/L 486DX-33 Color AMS TravelPro/L 486DX2-50 Color AMS TravelPro/L 486DX2-66 Color AMS TravelPro/L 486SX-25 Color	Associates Mega Sub-Systems, Inc.	800-886-2671; 818-814-8851	818-814-0782
PenExec Power Exec 386SL/25 Power Exec 386SL/25 Color Power Exec 386SL/25 ColorPlus Power Exec 386SL/25 EL Power Exec 386SL/25 EL Color Power Exec 486SL/25 Power Exec 486SL/25 Color Power Exec 486SL/25 ColorPlus	AST Research Inc.	800-876-4AST; 714-727-4141	714-727-9355
Vista PC-Tablet	Bel-Air Technologies, Inc.	408-720-4424	408-720-8807
Innova 486NX NoteJet 486	Canon Computer Systems, Inc.（subsidiary of Canon, Inc.）	800-848-4123; 714-438-3000	714-438-3099
Amiga 1200	Commodore Business Machines, Inc.（Computer Systems Division）	800-66-AMIGA; 215-431-9100	
CompuAdd 425tx（※） CompuAdd 425txt	CompuAdd Corp.	800-272-2843; 512-250-1489	512-331-6236
Compudyne 486SL/25 Notebook Compudyne 486SLC/25 Notebook	Compudyne, Inc.（subsidiary of CompUSA）	800-932-2667; 214-888-5700	214-702-0300
Walkabout/386SL	Data General Corp.	800-328-2436; 508-366-8911	508-366-1319
Dauphin 5000-C Color Pentop	Dauphin Technology, Inc.	800-782-7922; 708-971-3400	708-971-8443
BusinessStarter Notebook Dell 320SLi（※） Dell 325SLi	Dell Computer Corp.	800-289-3355; 512-338-4400	512-343-3653
DECpc 325SL Color Notebook DECpc 325SL Notebook DECpc 325SL Notebook Plus DECpc 325SLC Color Notebook DECpc 425SL Notebook DECpc 425SLC Color Notebook	Digital Equipment Corp.（DEC）	508-493-5111	800-234-2298
DECpc 425SLC Premium Color Notebook	508-493-5111	800-234-2298	
ActionNote 4000 486SLC/25	Epson America, Inc.（subsidiary of Seiko Epson Corp.）	800-922-8911; 310-782-0770	310-782-5220
Palmtop PC	FOCUS Electronic Corp.（subsidiary of FOCUS Electronic Co., Ltd.）	800-826-6880; 909-468-5533	909-468-5525
325 Point Poqet PC Poqet PC Plus PoqetPad PoqetPad Plus	Fujitsu Personal Systems, Inc.（subsidiary of Fujitsu, Ltd.）	800-831-3183; 408-982-9500	408-496-0609
GRiD 4025N GRiD 4025NC GRiD Convertible GRiD Convertible GT GRiD PalmPAD	GRiD Systems Corp.（subsidiary of Tandy Corp.）	800-356-4743; 817-491-5200	817-491-5998
386 Palmtop	Hansen Information Technologies	800-821-9316; 916-921-0883	916-921-6620
HP 100LX HP 95LX HP Vectra XM 486DX/33 HP Vectra XM 486DX2/50 HP Vectra XM 486DX2/66 OmniBook 300	Hewlett-Packard Co.	800-752-0900; 800 387-3867（CD） 415-857-1501	800-333-1917
Hyundai Courier	Hyundai Electronics America（subsidiary of Hyundai Electronics Industries Co., Ltd.）	800-727-6972; 408-473-9200	408-894-9751
9075 PCradio ThinkPad 500 ThinkPad 710T ThinkPad 720 ThinkPad 720C	IBM（International Business Machines）	800-426-3333; 914-765-1900	

製品名	会社名	電話番号	FAX番号
Color Keynote Compact 486SLC-33 Keynote Compact 486SLC-33 Keynote Compact 486SX-25 Keynote Force 486DX-33 Keynote Force 486DX-33 Active Color	Keydata International, Inc.	800-486-4800; 908-755-0350	908-756-7359
WinBook	Micro Electronics (division of MEI/Micro Center, Inc.)	800-468-7502; 614-481-4417	800-448-0308
Elite 486SLC-33 Subnote	MidWest Micro	800-942-8080; 513-368-2309	513-368-2306
NCR 3130 NotePad NCR 3150 NCR 3170 Safari NSX/20	NCR Corp. (an AT&T Co.)	800-225-5627; 513-445-5000	513-445-4184
Ultralite Autograph UltraLite Versa 20 UltraLite Versa 20C UltraLite Versa 25 UltraLite Versa 25C UltraLite Versa 25CP UltraLite Versa 33 UltraLite Versa 33C	NEC Technologies, Inc. (subsidiary of NEC Corp.)	800-632-4636; 508-264-8000	508-264-8673
DES-Based Security Card	New Media Corp.	800-CARDS-4U; 714-453-0100	714-453-0114
PenView	Norand Corp.	800-553-5971; 319-369-3100	319-369-3453
Color Cruiser 325C-460 Presentation Notebook Color Cruiser 425SXC-460 Presentation Notebook Color Cruiser 433DXC-460 Presentation Notebook Cruiser 325-460 Presentation Notebook Cruiser 425SX-460 Presentation Notebook Cruiser 433DX-460 Presentation Notebook	NovaCorp International, Inc.	716-647-6510	716-647-6512
Quaderno	Olivetti Office USA	800-527-2960; 908-526-8200	908-526-8405
Polycorder 286LX	Omnidata International, Inc.	801-753-7760	801-753-6756
Omron Toucan	Omron Office Automation Products, Inc. (division of Omron Corp.)	800-362-4411; 408-727-1444	408-970-1149
SLic 4300/25	PC & C Research Corp.	800-843-1239; 805-484-1685	805-987-8088
Infolio	PI Systems Corp.	800-831-6161; 503-293-9585	503-293-9590
Sager NP433 486 Subnotebook Sager NP740 486SX-25 Color AM Sager NP740 486SX-25 Color PM Sager NP740 486SX-25 Mono Sager NP741 486SX-33 Color AM Sager NP741 486SX-33 Color PM Sager NP741 486SX-33 Mono(※) Sager NP743 486DX-33 Color AM Sager NP743 486DX-33 Color PM Sager NP743 486DX-33 Mono Sager NP746 486DX2-66 Color AM Sager NP746 486DX2-66 Color PM Sager NP746 486DX2-66 Mono Sager NP840 486SX-25 Color AM Sager NP840 486SX-25 Color PM Sager NP840 486SX-25 Mono Sager NP843 486DX-33 Color AM Sager NP843 486DX-33 Color PM Sager NP843 486DX-33 Mono Sager NP846 486DX2-66 Color AM Sager NP846 486DX2-66 Color PM Sager NP846 486DX2-66 Mono	Sager Modern Computer, Inc.	800-669-1624; 818-964-4849	818-964-2381
PenMaster	Samsung Electronics America, Inc. (Information Systems Division)	800-446-0262; 201-229-4000	201-229-4110
DynaVision IIIC 386SL-25 DynaVision IVC 486SX-25	Scenario, Inc.	800-468-1119; 617-965-6458	617-965-6460
PC-3000 Palm Executive PC-6681 PC-6781 PC-6785 PC-6881 PC-6891 PC-7700 PC-7750	Sharp Electronics Corp. (Information Systems Group)	800-BE-SHARP; 201-529-8200	201-529-9637
T100X Dynapad T1900 Satellite T1900C Satellite T4500 T4500C T4600 T4600C	Toshiba America Information Systems, Inc. (TAIS) (subsidiary of Toshiba Corp.)	800-334-3445; 714-583-3000	714-583-3645
Verxion NB8004	Transource Computers	800-486-3715; 602-997-8101	602-997-8886
SuperGen 1000 Notebook	UNISYS Corp.	800-874-8647; 215-986-4011	215-986-2312

製品名	会社名	電話番号	FAX番号
Z-Lite 320L	Zenith Data Systems Corp.（a Groupe Bull Co.）	800-553-0331; 708-808-5000	708-808-4434
ZEOS ColorNote 486 ZEOS Pocket PC	ZEOS International, Ltd.	800-423-5891; 612-623-9614	612-633-1325

付録1-2●PCMCIA FAXモデムカード

製品名	会社名	電話番号	FAX番号
Starcard 1442fx Data/Fax Modem Starcard 2400fx Data/Fax Modem Starcard 2442fx Data/Fax Modem Starcard 2442vx Data/Fax/Voice Modem	AMT International Industries, Inc.	714-375-0306	714-375-0317
PCMCIA Fax/Modem IPB-1414 PCMCIA Fax/Modem IPB-2496 PCMCIA Fax/Modem IPB-9696（※）	Apex Data, Inc.	800-841-2739; 510-803-2020	510-803-9388
KeepInTouch Card 3760 Series	AT&T Paradyne（subsidiary of AT&T）	800-482-3333; 813-530-2000	813-530-2103
Smart One 1442 PCMCIA Smart One 1442P	Best Data Products, Inc.	800-632-BEST; 818-773-9600	818-773-9619
Road Warrior	C Centennial, Inc.	800-535-3668; 508-670-0646	508-670-9025
Complus FaXXpress 1414 PCMCIA Fax/Modem Complus FaXXpress 942 PCMCIA Fax/Modem	ComPlus, Inc.	510-623-1000	510-623-1004
PCMCIA Fax/Modem	Computer Products Plus, Inc.	800-274-4277; 714-847-1799	714-848-6850
RediCARD V.22bis RC-2496 RediCARD V.32 RC-9696 RediCARD V.32bis RC-1496	DATA RACE, Inc.	800-329-7223; 210-558-1900	210-558-1929
Credit Card Fax/Modem 2496CFM PCMCIA Fax/Modem	DataTrek Corp.	219-522-8000	219-522-0822
Fury Card 2400	Dr. Neuhaus	408-685-0928	
ThinFax 9624	EXP Computer, Inc.	516-496-3703	516-496-2914
FM144/144V（P） FM9648/24（P） FM9696/96V（P） SM24（P）	GVC Technologies, Inc.	800-289-4821; 201-579-3630	201-579-2702
OPTIMA 24 + FAX96 for PCMCIA	Hayes Microcomputer Products, Inc.	800-96-HAYES 800-665-1259（CD） 404-840-9200	404-441-1238
PCMCIA 96/24 Data/FAX Modem PCMCIA Data/FAX Modem	IBM（International Business Machines）	800-426-3333; 914-765-1900	
24/96 Mercury	Magic RAM	213-413-9999	213-413-0828
PCMCIA Fax/Modem CC3144 PCMCIA Fax/Modem CC324FM PCMCIA Fax/Modem XJ1144 with XJACK PCMCIA Fax/Modem XJ124FM with XJACK PCMCIA Fax/Modem XJ196FM with XJACK PCMCIA Fax/Modem XJ2144 with XJACK PCMCIA Fax/Modem XJ224 with XJACK	Megahertz Corp.	800-527-8677; 801-272-6000	801-272-6077
MultiModem LT	Multi-Tech Systems, Inc.	800-328-9717; 612-785-3500	612-785-9874
Info Blitz PalmModem	New Media Corp.	800-CARDS-4U; 714-453-0100	714-453-0114
PDPCMCIA-V.29	Pure Data, Inc.	214-242-2040	214-242-9487
SmartTalk 1414 SmartTalk 9624 SmartTalk 9624E	Smart Modular Technologies, Inc.	800-536-1231; 510-623-1231	510-623-1434
SupraFAXModem V.32bis PCMCIA DF2496A T24P/F Data/Fax Modem	Supra Corp. TDK Systems Development Center Toshiba America Information Systems, Inc.（TAIS）（subsidiary of Toshiba Corp.）	800-727-8417; 503-967-2400 916-265-5395 800-334-3445; 714-583-3000	503-967-2401 916-478-8290 714-583-3645
TWINCOM PCMCIA Optimum Card TWINCOM PCMCIA Platinum Card	TWINCOM（division of Target Technologies, Inc.）	800-666-2496; 919-395-6100	919-395-6108
WorldPort 14,400 Fax/Data PCMCIA WorldPort 2496 Fax/Data PCMCIA	U.S. Robotics, Inc.	800-342-5877; 708-982-5010	708-982-5235

付録1-3●PCMCIA ハードディスクカード

製品名	会社名	電話番号	FAX番号
Au126 Au63 Au63-III	Aura Associates	408-252-2872	408-252-2876
Derringer CP-1034	Conner Peripherals, Inc.	800-426-6637; 408-456-4500	408-456-4501
Cobra 1882PA PocketFile 42 PocketFile 85 Ranger 1841PA	Integral Peripherals, Inc.	303-449-8009	303-449-8089
MXL-105-III	Maxtor Corp.	800-4-MAXTOR; 408-432-1700	408-432-4510
MiniPORT 42P MiniPORT 64P MiniPORT 85P	MiniStor Peripherals Corp.	408-943-0165	408-434-0784
Caviar Ultra Lite CU140	Western Digital Corp.	800-832-4778; 714-932-5000	714-660-4909

付録1-4●PCMCIA LANカード

製品名	会社名	電話番号	FAX番号
EP-200 EP-200B(※) EP-200T	Addtron Technology Co., Ltd.	800-998-4638; 510-770-0120	510-770-0171
PCMCIA Thin Coax/UTP Ethernet Adapter(※)	Advanced Interlink Corp.	714-894-1675	714-893-1546
Portable PCMCIA Ethernet Adapter	American Megatrends, Inc.	800-828-9264; 404-263-8181	404-263-9381
StarLAN Ethernet Adapter	AMT International Industries, Inc.	714-375-0306	714-375-0317
PCMCIA Ethernet Card	C Centennial, Inc.	800-535-3668; 508-670-0646	508-670-9025
3270 MicroLink 5250 MicroLink	Capstone Technology	510-438-3500	510-651-7035
ENET-PCMCIA	Compex, Inc.	714-630-7302	714-630-6521
Credit Card Ethernet Adapter Model DE-650	D-Link Systems, Inc.	800-326-1688; 714-455-1688	714-455-2521
RediCARD Ethernet Adapter RC-10B2; RC-10BT	DATA RACE, Inc.	800-329-7223; 210-558-1900	210-558-1929
PCMCIA Ethernet Adapter	DataTrek Corp.	219-522-8000	219-522-0822
NE200T	Eagle Technology (unit of Anthem Electronics, Inc.)	800-733-2453; 408-441-7453	408-436-0348
G/LapCard	Gateway Communications, Inc.	800-367-6555; 714-553-1555	714-553-1616
Ethernet Credit Card Adapter Token Ring 16/4 Credit Card Adapter	IBM (International Business Machines)	800-426-3333; 914-765-1900	
TP-PCMCIA/C(※) TP-PCMCIA/T	Intellicom, Inc.	800-992-2882; 818-407-3900	818-882-2404
Genius LAN Model GE-3000	KYE International Corp.	800-456-7593; 909-923-3510	909-923-1469
LCEther PCMCIA BNC LCEther PCMCIA UTP	Link Computer, Inc.	714-993-0800	714-993-0705
PCMCIA Ethernet Adapter	Linksys (division of D.E.W. International Corp.)	800-326-7114; 714-261-1288	714-261-8868
Ethernet Adapter Token Ring Adapter	Magic RAM	213-413-9999	213-413-0828
PCMCIA 10Base2 Ethernet Adapter PCMCIA 10BaseT Ethernet Adapter	New Media Corp. New Media Corp.	800-CARDS-4U; 714-453-0100 800-CARDS-4U; 714-453-0100	714-453-0114 714-453-0114
PDMCIA9023-C; -T PDMCIA9025-4/16	Pure Data, Inc.	214-242-2040	214-242-9487
Socket EA Socket EA+	Socket Communications, Inc. Socket Communications, Inc.	510-670-0300 510-670-0300	510-670-0333 510-670-0333
Ethernet 10Base2 TC-Card Model TC5041-2 Ethernet 10BaseT TC-Card Model TC5041-T Token Ring 16/4 TC-Card TC4041-STP	Thomas-Conrad Corp.	800-332-8683 800-654-3822 (CD) 512-836-1935	512-836-2840
CreditCard Ethernet Adapter CE-10BC CreditCard Ethernet Adapter CE-10BT; CE-10B2(※)	Xircom, Inc.	800-438-9472; 818-878-7600	818-878-7630

付録1-5●その他のPCMCIAカード

製品名	会社名	電話番号	FAX番号
3270 Emulation Credit Card Adapter	IBM (International Business Machines)	800-426-3333; 914-765-1900	
Tempo	MediaShare Corp.	619-931-7171	619-431-5752
Bus Toaster Visual Media PCMCIA SCSI Adapter	New Media Corp.	800-CARDS-4U; 714-453-0100	714-453-0114
Serial I/O Credit Card Adapter	Socket Communications, Inc.	510-670-0300	510-670-0333
DES Security System	Techmatics	703-802-8300	

4

香港・ABCコンピュータ，リチャード・ロバート氏に聞く

「モービルコンピューティングは香港で，そして日本でも急速に普及する」

●著者●
川村研一
Kenichi KAWAMURA

●概要●
これまで「香港電脳街探訪指南」ほかで，ABCコンピュータ社のBICOM，HANDi-PCをレポートしてきたが，今回，同社のBIOSエンジニア，リチャード・ロバート氏に話を伺うことができた．開発者の考える次代の『サブノート』はどのようなものなのだろうか．



HANDi-PCの386版といえるBSL(BICOM386SL)の前には，286版のBICOM B260iが存在していた(本誌'93年1月号拙稿「香港電脳界探訪指南」参照)．BICOM386と同じボディサイズにIntel80286，当時としてはノートタイプのハードディスクの標準であった40MB，メインメモリ2MBを装備した上，先進的なPCMCIAスロットを標準で装備してあった．

B260iを開発したABCコンピュータ社でのコンセプトは，「軽量でポータビリティに優れ，しかも機能面でデスクトップに遜色のないもの」．上位機種に当たるBSLも，このコンセプトを引き継いだマシンである．CPUが386SLとなったのは，Windowsを動かすためにはせめて「386SX以上のプロセッサを」という，ごく当然の要求からだ．

BSLでは，ハードディスクの容量が60MB(最終的には80MB=BSL80)と50%アップ，メインメモリも4MBにアップしたが，PCMCIAスロットは，スペースに余裕がなかったため，外されてしまった．たった1kgのこのマシンでも，日本語Windows3.1上での基本的な日常業務にはまったく問題はない．

しかしながら，せっかくコンパクトにまとまったBSL80も，基本的な設計をB260iから流用しているため，(80286では何の問題もなかった)バッテリー容量(単3型電池5本)がネックになってしまった．386SLの高速CPUに加えて，メインメモリが2MBアップの4MBに，しかもハードディスクも80MBに容量アップとなった最終バージョンでは，バッテリーの力不足から「もっと長時間動く電池がほしい」との希望が，ユーザから出された．486SLマシン(HANDi-PC)の設計が，386マシンからのフィードバックを踏まえてのものとなったことはいうまでもない．

一番の焦点は，表示部分をCGA(640×400)から，VGAバックライト付き(640×480)としたこと．さらに，電池での使用時間を2時間から5時間と大幅に改善した．BSLでは外されてしまったPCMCIAのスロットも復活した．Windowsユーザの要望に答えたトラックボールの標準装備．キーボードのサイズも50%以上大きくなり，タイピングしやすくなった．また，PCMCIAスロット(TYPEⅡ×1)も装備されているので，名刺サイズのFAXモデムカードを利用すれば，持ち運びにまったくス

トレスを感じることなく，オフィスの環境をそのまま鞄に入れて持ち歩ける．レジュームアンドサスペンド機能が有効なことは，いうまでもあるまい．重量が1,5kgと，BSLの1.0kgに対して500g増加になったのは，これだけの機能アップを考えれば，やむを得ないだろう．

HPの100LXやOmniBook300など，英語環境ではものすごいマシンが出ているというのに，日本語環境ではまだまだという感じである．最近，話題になっているC.F.CのWin/Vが出れば，香港の日本人ユーザは非常にうれしいのだが．

●今後のHANDi-PCはどうなるのか？

今後，HANDi-PCはどのようになっていくだろう？まず予想されるのは，ハードディスクの大容量化．80MBでは足りないので，120MB，170MBになるだろう．CPUも486SL/25MHzから，486SL/33MHzに変わっていくだろう．

このような大容量化，高速処理化に進んでいくのは当然だと思うが，メーカーのABCコンピュータ社では，CyrixのCx486SLC(クロック未定)を搭載して価格を抑えたモデル“B3”，カラー表示の“B4”モデルも企画検討中である(現行のHANDi-PCはB2モデル)．

このあたりの事情を含めて，ABC社のBIOSエンジニア，リチャード・ロバート氏に話を聞いてみた．

川「もっと面白いコンピュータ出てこないかな?」

ロ「例えば?」

川「オートバイのヘルメットのバイザー部分に表示したりするの」

ロ「そいつはいいね．腕にキーボードをバンドで固定すれば，コーナリング中でもデバッグできる」

川「そうすると，キーボードはやっぱり右の腕に固定だね」

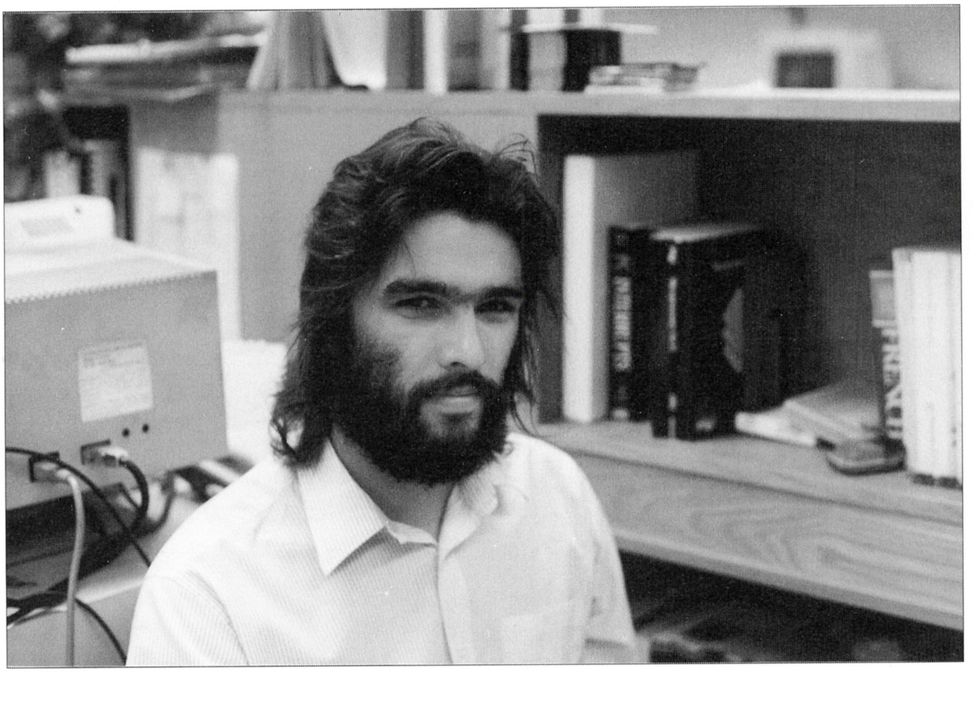
ロ「当然そうなるよ，右でアクセルもブレーキも操作するからね」

川「ヘルメットの上に後ろ向きで，CCDカメラをつけておいて，画像をバイザーに表示するのは，実用的でしょ．物体が近づいたらワーニング出すとか」

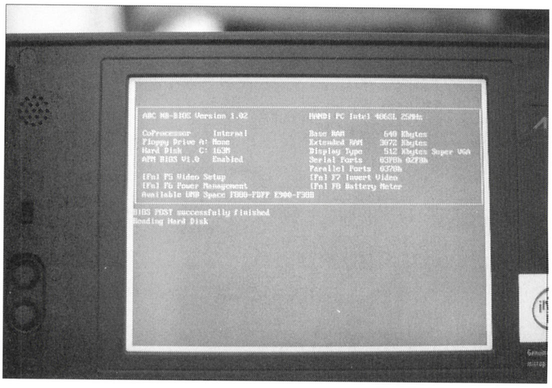
ロ「ポルシェなんか研究してるかもしれないけど，バックミラーやめてモニタを使うの」

川「ところでいま，どんなコンピュータがほしい?」

ロ「いまは，通勤に時間がかかるので，電車の中で使えるコンピュータがほしい．イメージとしては，モニタは，サングラスタイプで，最短距離で視点が合うようなもの．左の腕にキーボードをつけて，PCの本体が40MBのPCMCIAのハードディスクになっていて，総



◀Richard Roberts氏．'65年月3生まれの現在28歳．ロンドン動物園，ブリティッシュテレコムなどに勤務した後，「香港の方が面白そうだ」という理由で香港にやってきた．パワーマネージメントが優れていると評判のHANDi-PCのABC NB-BIOSは彼が書いたものである．その風貌から(?)，ニックネームは「ジーザス・クライスト」



▲HANDi-PCに搭載されているABC NB-BIOS Ver.1.02

重量は500gくらいで，3時間程度動くものがいい」

川「ターミネーターの世界だね」

ロ「ターミネーターじゃなくて，プレディターといってほしいね．込み合った電車の中でコンピュータを動かすには，座らなければいけないと思うでしょ．僕はだいたい座れないから，立ったまま使えることが前提となるんだ．日本の電車事情だって，TVで見る限りじゃひどい混雑じゃないか．通勤時間を利用して，コンピューティングするなら，立ったまま使えないと意味がない」

川「それ，非常に核心をついているよ」

ロ「僕は英語の環境でいいのだけど，日本のマーケットにはどうだろうね?」

川「漢字の表示が問題だね．ところで，B2(HANDi-PC)は日本のマーケットでも結構，評判がよいのだけど，使い方は今後どんなふうになるのだろうか?」

ロ「月並みな言い方だけど，移動中にコンピュータを通信端末として使うのがもっと一般的になると思う．例えば香港では携帯電話の普及がすごいでしょ．現にKENのところ(編集部注:川村氏が代表を務めるINOAC Software社のこと)でも，営業マン全員が携帯電話もっているでしょ」

川「携帯電話と電子手帳は，営業の三種の神器だから」

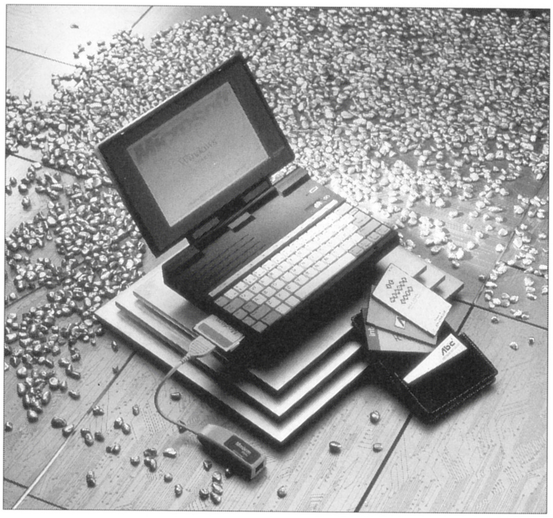
ロ「現在のところ，デジタル通信はまだできないけど，デジタルになったら，営業のスタッフにB2を持たせて，携帯電話でオフィスにデータを送信させたり，必要なデータを，客先で参照したりするのが可能になるでしょ」

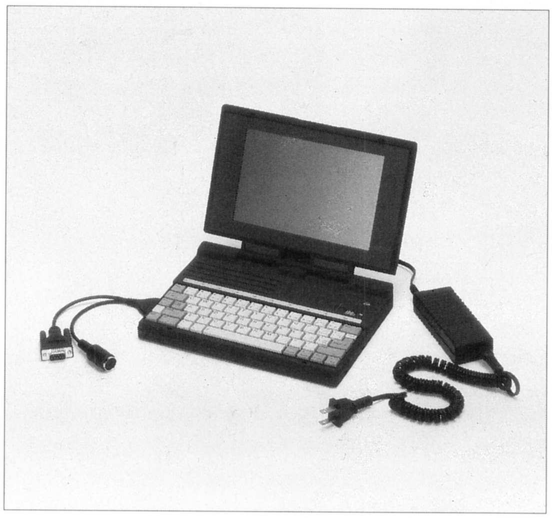
川「そうなると，もっと効率よく仕事ができる」

ロ「基本的にコンピュータは仕事の時間を短縮するためにあるのだから，その方面に利用するのがポイントでしょうな．香港での仕事には時間が勝負みたいなところがあるでしょう．だから，モービルコンピューティング(Mobile Computing)は，ものすごい勢いで普及すると思います．これは日本でも同じと思いますがね」

川「日本でのコンピュータ普及のポイントは，入力方法ではないかと思いますが．キーボードを使うのが嫌だから，コンピュータを使いたくないという人が大勢いますからね」

ロ「ペン入力方法はいろいろ試されているけど，一





度キーボードを使いなれると，ペン入力ではかったるく思えてくる．近いうちに僕も，ペン入力ボードロジックにチャレンジしてみたい．変換率などを高性能にするのは，結構自信あるよ」

川「ひょっとして，B2の後継バージョンは，ペン対応型になるのかな」

ロ「現状での研究は，いろいろやっているけどね．もっとも，いますぐにやるべきことはたくさんあってね．まず，CyrixのCx486SLC搭載の廉価版のB3でしょ．Intelの486SL/33MHz搭載のB2アッパーバージョン，それのカラー対応のB4．また，170MBハードディスク搭載の場合の電池消費率の試験．バッテリーチャージャのインテリジェンス化(できるだけ短い時間でチャージする)．ユーザの要求はきりがないからね」

川「386SL搭載のBSL80の電池が，もう少しもってほしいという私の要求はかなうのだろうか?」

ロ「ウォークマンの薄べったい電池を使って，ボルテージを現状の6Vから7.2Vにすれば，実質の使用可能時間は約2倍になるから，KEN，頑張って改造したら」

川「例えばさっきの話，386SLをCyrixの486SLCに載せ代えて，170MBのハードディスクを組み込んで，電池を7.2Vにして，1kgくらいにまとめたら，ポータビリティ抜群のマシンができると思うけど?」

ロ「面白そうだね! KENの会社で，超小型コンピュータを開発するのなら，協力するよ」

川「今日は時間を取ってもらってありがとう! 最後に，ざべの読者にメッセージをお願いします」

ロ「久しぶりに面白い話をしたね．こちらこそありがとう．ABC社は今後もどんどん小型コンピュータに力を入れて頑張るから，応援してほしい．それから，日本にまた連れってくれよ．あの魚(さば寿司)の味が忘れられない！」

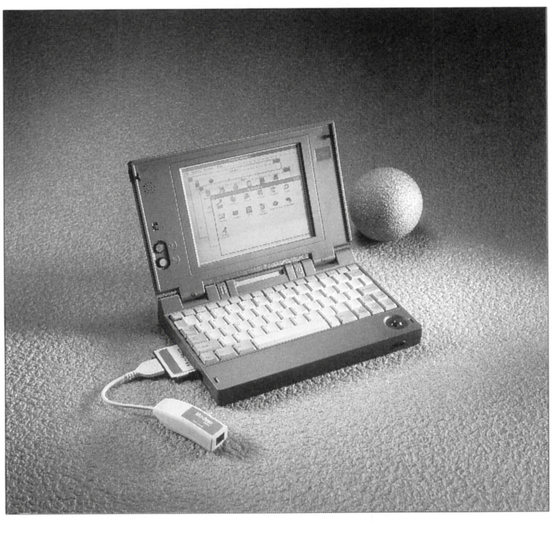
●求められるトータルコーディネーション

BSL80のバッテリーを長持ちさせる方法を現在，INOACでは真剣に考えている．ラジコン用の7.2VバッテリーをA/Cアダプターに直接装着してみたところ，約5時間動作した．いま実験しているのは，ウォークマンの薄べったい電池をBSL80の裏に貼り付けて，どの程度の時間持つのかを実験中である．この実験がうまくゆけば，BSL80の厚みを6mm程度増やして，電池容量を増やして，バッテリー駆動時間を現状の1時間から，せめて3時間くらいに延ばしたい．

最後に，ノートブックやサブノートのメーカーが付属販売しているキャリーケースはいただけない．せっかくコンピュータ自体は軽いのに，バックの取っ手の形状が悪く，重く感じてしまう．使い勝手が悪い．キャリーケースなどのアクセサリも，真剣に考えるべきではないであろうか．

これからのサブノートには，使い勝手のよい，本格的なアクセサリ，例えば高速チャージャとか，車の中のシガーライターからの充電，各国の電話回線につなぐことができるような電話アダプタ，軽量なFAXモデム，そしてそれらのアクセサリを効率よく収納できるセンスのよいバック．こうしたトータルコーディネーションが必要になってくると思う．

コンピュータだけ開発しているだけではだめになってしまう時代がすぐそこまで来ている．



◀ABCコンピュータのサブノートシリーズ．左からB240i，B386SL80，HANDi-PC B2-4170．B240iはB260iのHDD40MB版．B240iで装備されていたPCMCIAスロットがBSLシリーズでは外されてしまったのは，やむを得ないだろうが，残念．サブノートもHANDi-PCのように，いよいよ486+HDD170MBのスペックをもったマシンが出そろいつつある．携帯性だけでなく，速度面，記憶容量面でも，実用と呼ぶにたるレベルに達している．

特集・サブノートの気になるあのこと

5

サブノートPCレビュー

●各マシンのデータ表について●
・重量には，バッテリーを含みます．
・PCMCIAスロットで特にことわり書きのないものは，PCMCIA2.0準拠です．
・データは12月3日現在のものであり，その後変更される可能性があります．
●ベンチマークテストについて●
・WindSock3.30はChris Hewitt氏作のフリーソフトウェアです．nenpi.exeは広野忠敏氏作のフリーソフトウェアです．
・nenpi.exeは1分ごとにHDDに書き込みを行い，バッテリー駆動時間を計測するものです．バックライトONでの計測値です．
・3DBench，WindSockではSMARTDRVを組み込んであります．Windowsは386エンハンスドモード．
・CONFIG.SYSはHIMEM.SYS，SMARTDRVを組み込んだ以外，不要と思われるデバイスは外してあります．
・WindSockのMEMORYはALLを実行した後，MEMORYを何度もクリックして最大値を取ってあります．

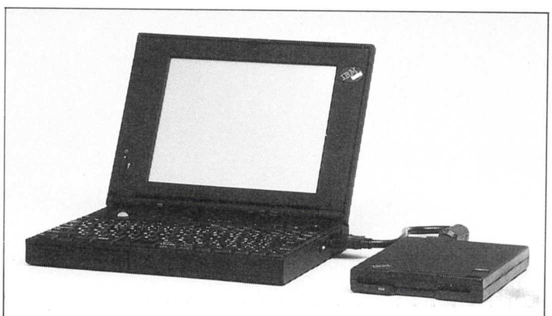
A5ファイルサイズ，1.0kg
サブノートブームの火つけ役

ThinkPad 220

日本アイ・ビー・エム

現在のサブノートブームの火つけ役となったマシンがIBMのThinkPad220だろう．A5ファイルサイズ，1kgのボディが一世を風靡したのは記憶に新しいところだ．ThinkPad220は，CPUにi386SL/16MHz，メモリは標準で2MB(最大6MB)，ハードディスクは80MBを搭載している．DOS J5.02/Vがプリインストールされているため，マシンを買ってすぐに使うことができる．ハードディスクの増設・交換はできない．

このスペックは，Windowsをエンハンスドモードで動作させるための最小スペックともいえるだろう．実際にこのマシンでWindowsを動作させるには勇気が必要だ．一応動作はするが，お世辞にも快適とはいい難い．スカスカとWindowsを使いたいのならば，486SLでメモリが4MB以上搭載されているものを選択すべきだろうが，Windowsにこだわらないのならば，ThinkPad220は十分実用的なスペックを持っているマシンだといえる．サブノートでWindowsベースのPIM*ソフトをスカスカ使うのも魅力的だが，ThinkPad220は，あくまでもDOSマシンとして使いたいサブノートだ．



▲ThinkPad 220

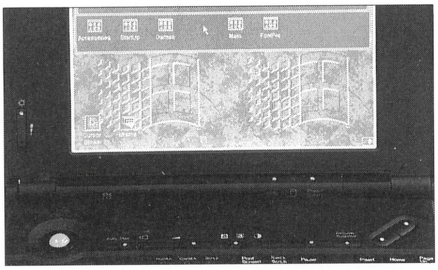
画面は7.7インチの16階調モノクロ液晶表示，液晶ディスプレイはやや見にくいが，画面の大きさは十分だ．FDDは1.2MBフォーマットのディスクも読める3モードドライブが外付けで用意されている．また，PCMCIA2.0/JEIDA4.1準拠のTYPEⅡスロットが1基用意されている．PCMCIA TYPEⅡスロットにはフラッシュメモリやFAXモデムカードなどを装着することが可能だ．インターフェースはシリアル×1，パラレル×1のほかに，キーボードインターフェースとVGAインターフェースが用意されている．VGAモニタを使って，外部モニタに表示させることもできるが，必要性はさほどないかもしれない．電源はACアダプタと単3乾電池6本で駆動される．ACアダプタは持ち歩くことを前提に考えてか，非常にコンパクトに設計されている．コンセントの部分は，ACアダプタ本体に格納することができるため，持ち運びには便利だ．

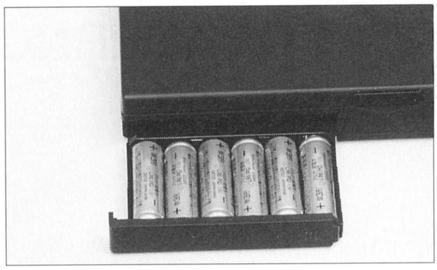
●A4サイズのThinkPadをそのまま小さくした外観

外観は，他のThinkPadシリーズど同様の黒い弁当箱スタイルで，ちょうどA4サイズのThinkPadをそのままサイズダウンした感じだ．キーボードもA5サイズに収めているため，小さい．実際に，ワープロソフトを使ってキーボードの感触を試してみた．キーの大きさはさすがに小さいが，なかなか使えるという感触を受けた．さすがに，タッチタイプをするにはキーが小さいため少々コツが必要だが，実用上問題はないだろう．ただし，長時間の使用に耐えられるかどうかは疑問が残る．

キーボード上部には，トラックボールと各種インジケータが配置されている．トラックボールはキーボード左上，トラックボールの二つのボタンはキーボード

*Personal Information Manager


▲トラックボールとボタンは左右に分かれて配置


▲単3乾電池6個で動作する

の右上に配置されている．机の上にThinkPad220を置いて，片手でトラックボールとボタンを使うわけにはいかないが，両手で操作するには非常に楽だ．たとえば膝の上で操作するときや，本体を固定せずに立って操作するときも楽に使うことができる．反面，キーボードとトラックボールを併用して操作するにはちょっとしたコツが必要になるだろう．

●ノートメニューで楽々

環境の構築は，あらかじめDOS/Vが内蔵ハードディスクにインストールされているため，あまり問題はない．DOS/V初心者でもマシンをセットアップすることなく，簡単使うことができる．ハードディスクにはほかのThinkPadシリーズと同様にノートメニューがインストールされているため，初心者でもある程度は使いこなすことが可能だ．サポートについても，本家IBMのサポートを受けることができるというのは心強い．

実際にこのマシンを使ってみると，ThinkPadは微妙なバランスを保ったマシンだということを感じる．すなわち，さほど使いにくいマシンではないが，これ以上スペックを落とすと使いものにならない．つまり，実用上ぎりぎりのスペックを満足したマシンであるということだ．キーボードが使いにくい．CPUがちょっと遅い．など不満はある（ひょっとしたらThinkPad220は単なるプローブとしてマーケットの動向を探るためにIBMが作った戦略的な商品ではないのだろうか）が，A5ファイルサイズ，重さ1kgのメリットは大きい．しかし，これらの不満を我慢できる，継続的に持ち歩ける程度の実用的な軽いマシンがほしい．そんな人にThinkPad220はうってつけのマシンだといえるかもしれない．

●乾電池駆動の問題

このマシンの最大の特徴であり問題であるもいえる点が「乾電池駆動」だ．カタログスペックでは，アルカリ単3乾電池で1～8時間の連続駆動が可能とあるが，実際にはそんなには動作しないようだ（本誌'93年10月号「ThinkPad220の期待と誤算」を参照されたい）．実際に使ってみるとアルカリ乾電池では1時間～2時間程度しか駆動できないようだ．アルカリ単3乾電池6本の値段もバカにならないため，かなりランニングコストがかかる計算になる．なぜ最初からバッテリー駆動にしなかったのだろうか．とはいえ，乾電池で動作することのメリットは大きい．NiCdバッテリーなどを採用しているサブノートでは，バッテリーがなくなったらAC電源を探して使うか，バッテリーに充電するかしなければ，マシンを使うことはできないが，乾電池駆動であれば，近くのコンビニで電池さえ買ってくれば，いつでも使えるのだ．

しかし，ACアダプタと単3電池による電源だけではなく，充電して再利用可能なNiCdバッテリーなどを標準装備すれば，ThinkPad220のフィールドはさらに広がるだろう．NiCdバッテリーなどでも動作すれば，普段はAC電源駆動，持ち運ぶときはバッテリー駆動，緊急時には乾電池駆動というようにTPOにあわせて使うことができるハズである．もし，ThinkPad220の後継機が発売されるなら，電源はAC，バッテリー，乾電池の3WAYにしてほしいところだ．

いろいろ書いたが，A5サイズで軽いマシンがほしいというのなら，ThinkPad220は買いだろう．（広野忠敏）

●ThinkPad 220データ

・CPU	i386SL/16MHz
・メモリ	2MB（最大6MB）
・HDD	80MB
・FDD	外付け標準装備
・ディスプレイ	16階調モノクロ液晶，640×480
・PCMCIAスロット	TYPE II ×1
・電源	ACアダプタ，単3アルカリ乾電池6本
・外形寸法・重量	226（W）×166（D）×32（H）mm，1.0kg
・その他	IBM DOS J5.02/Vプリインストール
・価格	248,000円（標準価格）
・問い合わせ先	ダイヤルIBM／フリーダイヤル0120-1-55307

●ベンチマークテスト結果

・WindSock3.30	
CPU：55／VIDEO：35／DISK：58／MEMORY：71／ALL：61	
・3DBench	4.9
・nenpi.exe	（バックライトOFF時で2:00前後．くわしくは本誌'93年10月号をご参照ください）

デザインのかっこよさ，サンプリング機能，使いやすいOrganizerと魅力一杯の個性的なサブノート

QUADERNO 33

日本オリベッティ

日本オリベッティより発売されたサブノートパソコン「QUADERNO 33」は非常に個性的である．他のサブノートパソコンに比べ，「道具」としての色が濃いような感じすらする．イタリアのデザイナー，マリオ・ベリーニ氏の手によるデザインで，表面の凸凹や側面のスピーカー部分などがかわいい仕上がりになっている．ジャストA5サイズと非常にコンパクトでありながら，テンキーやトラックボールの工夫された配置によって操作性が確保されている．

QUADERNO 33は，AMD社の3.3V駆動の省電力型CPU，AMD386SXLV/20MHzを採用している．RAMは標準で4MB，増設はPCMCIA準拠のICカードスロットで行う．フラッシュROMが採用されていて，BIOSや内蔵のROM DOSのアップグレードが可能になっている．内蔵のROM DOSは残念ながら英語DOS Ver.5であるが，付属のFDに入っているユーティリティを使って日本語化することも可能だ．内蔵HDDの容量は60MBで交換はできない．ディスプレイはバックライト付き白黒のSTN型液晶でムラがとても少なく，非常に視認性が高い．画面の大きさは7インチ(140.8×105.6mm)で，モノクロ16階調の表現ができる．

気になる重量は1.356kg(バッテリー込み)と，現在のサブノートパソコンの中でも軽い部類だろう．バッテリーは標準で7.2V 1700mAのNiCd充電パックが付属する．フル充電で2時間以上の連続動作が可能だ．本体の大きさは210×148×42mmと小さいが，少々厚みが気になる．キートップが10mm四方で，キーとキーのピッチは12mm程度，成人男子がホームポジションに指を綺麗に揃えてキー入力するのは物理的に難がある，キーストロークは1.2mmと少ないが，キートップがスムーズに沈むために感触は悪くない．サブノートパソコンにはめずらしくテンキーがある．本体右側にあるトラックボールは直径11mmのボールが使われていて慣れるとポインティングもスムーズに行える．ボタンは本体上面に2個，右側面端に1個と3ボタン方式で，親指でボールを操作しながら人差し指は右側面のボタンを使うと快適な操作を行うことができた．バッテリーの残容量，CPU・キーボードなどの本体の状態や現在時刻がディスプレイの付け根に用意された小さな液晶で確認できる．



▲QUADERNO 33　キーボード左上に10キーが配置されている

本体右側面にはPCMCIA2.0準拠のICカードスロット(TYPEⅡ)が一つ用意されている．メモリ増設のときにもこのコネクタを利用することになっているので，LANカードやカード型モデムを使う場合にはメモリ拡張はあきらめないといけない．各種インターフェースコネクタは本体後部側面にまとめられている．すべてのコネクタは特殊なもので，必ず専用のケーブルが必要となるため，外出先で何かの機器と接続する場合には専用ケーブルを必ず携帯しなくてはいけないことになってしまう．

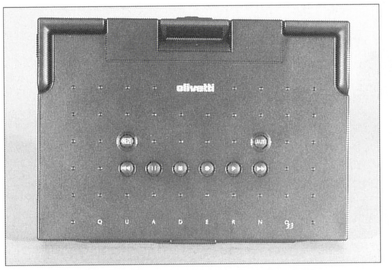
本体を見てまず気づくのは，蓋の上面にあるカセットデッキのようなスイッチだ．これは内蔵のマイクを使い，音声をサンプリングするときに操作するもので，カセットテープに録音するような感覚で，音声をデータ化し，内蔵HDDに記録することができる．もちろん記録した音声データはサブノートとは思えないほどの立派な内蔵スピーカーを使って再生することができる．マイク端子やスピーカの出力端子も用意されていて，クォリティを要求される場合も大丈夫だ．サンプリングの方法はVoice Recorder，Sound Recoderの2種類を選択できる．Voice Recorderでのサンプリングレートは8kHzで，サンプリングと同時にデータを圧縮してHDDに記録していく．1秒の音声をサンプリングすると圧縮方法によって，11KB/4KB/1.6KBのデータが作られる．一番圧縮率の高い方法で1時間の会議をサンプリングした場合，5.76MB

のデータファイルが後に残る計算だ．もちろん圧縮率の高いものを使うと音質も当然落ることになる．一番圧縮率の高いGSM形式だと，会話の内容がなんとか判断できる程度の音質になってしまう．一方，Sound Recoderを使うと11.025kHzのレートでサンプリングが可能でAMラジオ程度の音質は確保される．これらの機能はすべてWindows上で動作し，液晶パネルを閉じたままの状態でも蓋上のスイッチで簡単に起動できる．

QUADERNO 33には英語版Windows 3.1のほかに「Lotus Organizer for Olivetti」「Works for Windows」がアプリケーションとしてプリインストールされている．この二つのアプリケーションはWindows用で，FD形式では添付されていないので，何らかの手段でバックアップをとる必要があるだろう．バックアップをとる前に，何かの手違いで内蔵HDDをフォーマットしてしまったりするとQUADERNO 33の機能のほとんどが手元から消えてしまうということだ．また，紹介した音声サンプリング機能もWindows上で動作する．これらの付属ソフトウェアのマニュアルは一切付属せず，Windowsで用意されているオンラインヘルプを参照する以外の手段はない．このモデルを英語版として販売している理由の一つに日本語のドキュメントが整備されていないということがあげられる．

このようにQUADERNO 33は，Windows用のサブノートだと言ってもおかしくないだろう（QUADERNO 33は通常 MS-Windowsを起動しておき，使わないときにはレジュームしていつでもWindows環境に戻れるようにしておくのが正しい使い方だと思う）．もちろん，MS-DOSやDOS/Vを動作させることは十分可能であるが，それだけではQUADERNO 33特有の機能はほとんど生かせないことになる．

付属のWindows用アプリケーション「Lotus Organizer for Olivetti」は本来簡易OAツールなのだが，ここではQUADERNO 33をシステム手帳のように手軽に使える道具に変えてしまうツールと表現してもおかしくないだろう．QUADERNO 33でこのアプリケーションは非常に重要な役目をもっている．ひとつはOAツールとしての側面，いわゆるシステム手帳ソフトなのだ．そして，もうひとつは管理ツールとしての側面だ．



◀蓋上にカセットデッキのようなスイッチを使い，サンプリングができる．スピーカーは本体右側面に内蔵されている

QUADERNO 33の魅力は外見のデザインやサンプリング機能だけではなく，Lotus Organizerを中心とした身近な道具としての環境である．道具として非常に細かいところまで気遣いを感じさせる作りとなっている．DOS/V化して出先で文章を日本語入力するといった使い方もまったく不可能ではないが，コンパクトなキーボードのため快適にはいかないかもしれない．

いずれにせよ，日本語版QUADERNO 33が出ることを期待したい．（一ヶ谷兼乃）

●QUADERNO 33データ

CPU	AMD386SXLV/20MHz
メモリ	4MB
HDD	60MB
FDD	オプション
ディスプレイ	16階調モノクロ液晶，640×480
PCMCIAスロット	TYPE II ×1
電源	ACアダプタ，NiCd
外形寸法・重量	210（W）×148（D）×42（H）mm，1.35kg
その他	ROM DOS5.0，Windows3.1（ともに英語版），Lotus Organizer for Olivetti，Works for Windowsプリインストール
価格	223,000円
問い合わせ先	日本オリベッティ㈱PC販売部／℡0120-045-393

●ベンチマーク結果

・Windsock3.30
CPU：54／VIDEO：35／DISK：57／MEMORY：22／ALL：51
・3DBench　4.9
・nenpi.exe　2:23

HANDi-PC

本誌「香港電脳雑錦編」でもお馴染み
強力なパワーマネージメント機能

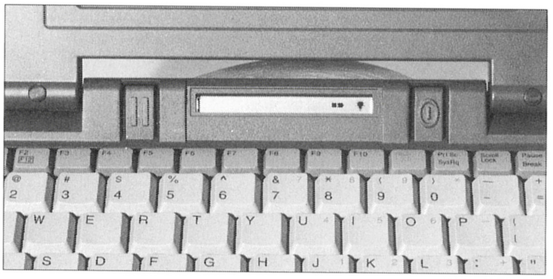
プロサイド

HANDi-PCは，今回レビューするサブノート群の中でも，その性能は上位にランクづけすることができる．

BIOSはABC BIOS，ちなみにこのマシンは香港のABC Computer社製だ．FDDは3.5インチ外付けがオプションで用意される．さらに，PCMCIA2.0 TYPE II のカードスロット×1と，サブノートとしては申し分ない



▲HANDi-PC



▲液晶インジケータは蓋を閉めても確認でき，使い勝手がよい．
キーボードは，次に紹介するZ-LITEとともに，サブノートとしてはキートップが大きく，タイピングしやすい

スペックだ．インターフェースはシリアル×1，パラレル×1のほかに，キーボードインターフェースとVGAインターフェースがそれぞれ用意されている．VGAインターフェースを介して外付けのディスプレイを接続すれば，800×600ドットのSVGA相当の解像度まで表示することができる．

外観を眺めてみると，液晶インジケータとキーボード右下のトラックボールが目につく．インジケータには，バッテリーインジケータやハードディスクのアクセスランプ，レジューム状態が表示される．このインジケータは蓋を閉めた状態でも確認することができるので，なかなか使い勝手がよい．トラックボールはキーボード右下にあり，ボタンはコーナーの手前と右側面にある．トラックボールを使う際には人指し指で，トラックボールを操作して親指でボタンを押すことになるが，慣れないとかなり使いにくい．トラックボールの左には，バッテリーが配置される．この位置はちょうどキーボードの下になる．そのため，バッテリー部分がパームレストになり，キーボードの操作は楽に行えるだろう．

●強力なパワーマネジメント機能

HANDi-PCの一番の特徴は，強力なパワーマネジメント機能にあるといってもよいだろう．ABC BIOSのパワーマネジメント機能は，ハードディスクやディスプレイ，シリアル/パラレルポートのアイドルタイムから，レジュームに移行する時間，レジュームから復帰するときのトリガなど，かなり細かく設定することができる．さらに，レジューム機能は状態をオンメモリに保存するのではなく，ハードディスクに保存して，レジュームから復帰するときはハードディスクに保存された状態を元に復帰させることも可能だ．

ちなみに，HANDi-PCではバッテリー駆動時にバッテリーが切れる直前になると，パワーマネジメントの設定にかかわらず，ハードディスクにレジューム状態を保存してから，電源が切れるようになっている．従来のこの手のマシンでは，バッテリがなくなりそうになると，あわててハードディスクに作業中の情報を保存するなどの手間が必要だったが，HANDi-PCでは作業中の情報を自動的に保存して，次回に起動したときは自動的にレジュームされるため安心だ．

気になる連続動作時間だが，ベンチマークの結果ではパワーマネジメントがOFFで，バックライトが常時点灯している状態で2時間20分をマークした．バックライトを点灯しないで，パワーマネジメントをONにすれば，さらに長時間の連続駆動が可能だろう．

HANDi-PCは値段・性能・大きさともに満足できる水準にある，数少ないサブノートであるといえる．Windowsを動作させてもさほどストレスを感じないし，パワーマネジメント機能をフルに活用すれば，TPOにあった使い方も可能だ．あとは，発売元のプロサイドのサポートに期待したい．しっかりしたサポートが期待できれば，現状ではベストな選択ではないだろうか．　（広野忠敏）

●HANDi-PC 486/B280データ

・CPU	i486SL/25MHz
・メモリ	4MB（最大16MB）
・HDD	80MB（*）
・FDD	オプション
・ディスプレイ	16階調モノクロ液晶，640×480
・PCMCIAスロット	TYPEⅡ×1
・電源	ACアダプタ，NiMH
・外形寸法・重量	255（W）×168.8（D）×36.2（H）mm，1.5kg
・価格	229,000円
・問い合わせ先	プロサイド㈱／℡03-3255-1517

●ベンチマークテスト結果

・Windsock3.30
CPU：187／VIDEO：110／DISK：143／MEMORY：186／ALL：151

・3DBench	18.1
・nenpi.exe	2:20

*現在，HDD170MBタイプの製品が出荷されている．価格は279,000円．

機能美にあふれたサブノート
トラックボール部分は取り外し可能

Z-LITE 425L Model 170W

日本データシステム

Z-LITE 425Lは機能美にあふれ，実にパワフルなサブノートコンピュータだ．外観は他のサブノートによく見られる無骨な箱型ではなく，スマートである．手元に向かって段々と薄くなっており，横からみるとくさび型になっている．後部側面も丸みを帯びており，高級感さえ漂ってくる．大きさはA5サイズよりひと回り大きい．この大きさがキーボードの余裕につながっている．デスクトップパソコンのキートップは13mm四方，19mmピッチで配置されているのだが，Z-LITEのキートップは12mm四方，16mmピッチのため，デスクトップパソコンを使っている感覚でキータイプが可能だ．キーの感触もよく，キーボードに関しては申し分ない．

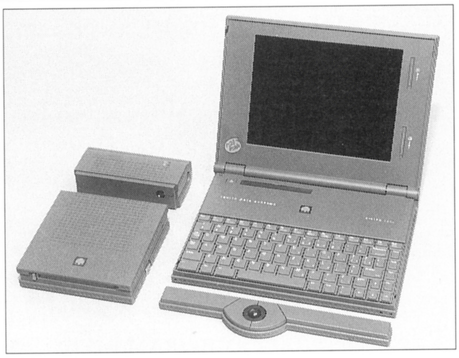
このモデルの特徴となっているのが「LITEPOINT」と呼ばれる取り外し可能なトラックボールである．キーボードの手前に取り付け，ホームポジションに指を置くとボールが親指で操作できる位置にくる．ボールの直径は16mmと比較的大きく，細かな操作もやりやすい．取り外しできるのは便利な反面，強度的な面で若干気になるところだ．

CPUは3.3V駆動のi486SL/25MHz，RAMは標準で4MB(最大8MB)．RAMの増設は本体裏に用意された専用コネクタを使用するために，ICカードのスロットが消費されることもない．BIOSはフラッシュROMになっていて，ソフトウェアで簡単にアップデートできる．ディスプレイは8.5インチSTNモノクロ液晶で，ムラも少なく，640×480の解像度で64階調表示を実現している．VRAMは256KBで，外部にモニタを接続すると解像度1,024×768/16色が可能だ．Windowsでは付属のディスプレイドライバを使用し，表示をパンニングして1,024×768ドットの仮想スクリーンを使用できる．

今回，試用したModel 170Wには170MB HDDが内蔵されている．内蔵バッテリーはNiMH(8.4V 2200mA)の大容量タイプのもので，フル充電で2時間30分を超える連続動作を実現している．重量はバッテリーを含んで1.77kgである．

PCMCIA2.0準拠TYPEⅡスロットが2個用意されている．

本体後部側面が丸くなっているため，XJACKのような形状のものは使えない場合が出てくる可能性がある．左側面にはACアダプタ(110V/220V)やFDDと接続するための非常に特殊形状のコネクタがある．付属のFDDとACアダプタの二つは合体し，本体同様のデザインになる．



▲Z-LITE

Z-LITEには，DOS6.0，Windows3.1(ともに英語版)，各種ドライバソフト，Windows上で動作させるオンラインマニュアルのFDが付属し，プリインストールされている．各ソフトウェアにはくわしく記述された英語版ドキュメントが付属しているので安心だ．CMOS関連から，液晶のON/OFF，内蔵スピーカーのON/OFF，CPUのFAST/SLOWなどのハードウェアに関するすべての設定がキーボード操作で自在にできるのは特筆に値する．

Z-LITEは正統派のパソコンだ．大容量のHDD，操作性のよいキーボードなど，Windows環境でもハードウェアのスペック不足を感じることはない．質感もよく，所有感をくすぐる，高級サブノートのお手本となるモデルである．ここまでよくできていると「あと300gくらい軽くならないかなぁ……」とつい欲張ってしまう．

(一ヶ谷兼乃)

●Z-LITE 425L Model 170Wデータ

・CPU	i486SL/25MHz
・メモリ	4MB(最大8MB)
・HDD	170MB(＊)
・FDD	外付け標準装備
・ディスプレイ	64階調モノクロ液晶，640×480
・PCMCIAスロット	TYPEⅡ×2
・電源	ACアダプタ(FDD共用)，NiMH
・外形寸法・重量	251(W)×193(D)×28～35.5(H)mm，1.77kg
・その他	DOS6.0，Windows3.1(ともに英語版)プリインストール，パンニングモード
・価格	368,000円(予価)
・問い合わせ先	日本データシステム㈱／☎06-415-2641

●ベンチマーク結果

・WindSock3.30	CPU：185／VIDEO：152／DISK：158／MEMORY：171／ALL：161
・3DBench	17.5
・nenpi.exe	2:46

＊HDD85MBタイプの製品もある．価格は298,000円．

日本語使用を前提としたサブノート
ドッキングステーションで拡張性も十分

486 SWIFT Sub-Notebook Computer

エスエスケーベル

SSK-BELLが発売しているSWIFTは，現在発売されている他社のサブノートと比べると若干大きめだ．しかし，i486SX/25MHzを心臓とし，80MBハードディスク，標準4MBのRAM，OADG日本語配列キーボードで武装し，キーボード右下にトラックボールを内蔵，外付けタイプFDD標準添付と，強力なCPUを中心に，まずは上位にランクされるスペックだろう．

今回評価したモデルには4MBのRAMが増設してあり，合計で8MBのRAMを搭載．下手なデスクトップマシンも顔負けというスペックである．実際，DOS環境では動作速度に不満はない．Windowsも速度面に関しては，十分実用に耐えうるものである．

拡張性も優れており，RAMは4MBもしくは16MBの拡張モジュールを増設することで，最大20MBまで搭載できる．FDD，9ピンシリアル，パラレル，外部キーボード，CRT出力といった標準的コネクタを装備．そして，標準でPCMCIA2.0(JEIDA4.1)準拠のICカードスロットも用意されており(TYPEⅡ×1)，モデムやLANアダプタの使用も容易となっている．

SWIFTには，Chips&Technologiesの65520というVGAチップが使用されている．液晶画面では640×480だが，この65520には512KBのVRAMが実装してあるので，外部CRTにつなぐことで，最大1,024×768の解像度まで表示可能だ．

拡張性に関して特筆すべきは，ATバスを内蔵していることである．別売のドッキングステーションと併用することで，市販のATバス用拡張カードを2枚まで内蔵することができる．

●日本語使用を考慮

A5サイズのサブノートが登場した現在，SWIFTは決して小さいとはいえない．しかし，キーボードを叩くことを考えると，このB5サイズは悪くない選択である．



▲SWIFT

キー下のパームレストといい，使い心地は快適だ．しかし，それでもキーを詰め込んだ感じは否めず，「¥@[:].//¥」の8キーの大きさが他のキーの半分程度しかない．「¥」を入力しようとして左隣の「^」まで押してしまうことがたびたびだ．というのも，キートップに日本語が表示されているOADG日本語配列仕様のキーボード(別にUSキータイプも用意されている)のためである．キーボードに関しては好みが別れるところだが，国内では標準的なOADG配列は日本語を使用する上では親切な設計である．

日本語といえば，マニュアルは完全日本語版である．万全なマニュアルとは言い難いが，SETUPからBIOSの設定まで日本語で説明されているので，初心者でも安心して使えるだろう．

SWIFTは初心者からヘビーユーザまで使える強力サブノートであるが，携帯性に不満がある．本体重量1.8kgは，一見たいした問題にはなりそうにないが，長時間持ち歩くとイヤになる．付属の外部FDDや，ACアダプタを加えると，2kgを超えるのではなかろうか．

連続使用時間にも不満を覚えた．Windowsを使用して，だいたい2時間程度動作する．Windowsを使用してであれば立派な時間といえるかもしれないが，はたして2時間程度しか動作しないものを持ち歩いてまで使用する必要があるのだろうか？　とはいえ，これはSWIFTにだけいえる問題ではない．サブノートという，携帯することを前提としたコンピュータに共通の課題である．

(佐伯英和)

●SWIFT(モデル1/日本語キーボードタイプ)データ

・CPU	i486SX/25MHz
・メモリ	4MB(最大20MB)
・HDD	80MB(着脱式)(＊)
・FDD	外付け標準装備
・ディスプレイ	64階調モノクロ液晶，640×480
・PCMCIAスロット	TYPEⅡ×1
・電源	ACアダプタ，NiCd
・外形寸法・重量	282(W)×180(D)×37(H)mm，1.8kg
・価格	258,000円
・問い合わせ先	㈱鈴木製作所 エスエスケーベル コンピュータ事業部／℡0292-85-8956

●ベンチマークテスト結果

・WindSock3.30
CPU：185／VIDEO：117／DISK：153／MEMORY：199／ALL：168

・3DBench	17.8
・nenpi.exe	2:31:00

＊HDD120MBタイプの製品もある．価格は日本語キーボードタイプで288,000円．なお，それぞれ英語キーボードタイプの製品もある．

バッテリー駆動時間の長さと
PCMCIA2.0 TYPEⅢがウリ

GemPower486S/25

クラスター

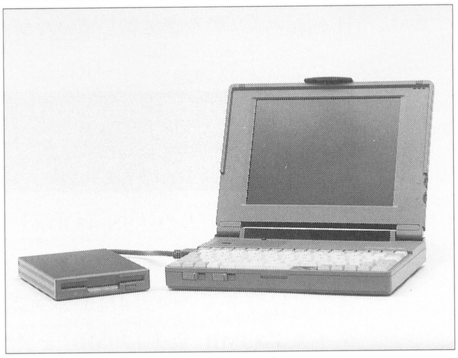
GemPowerはクラスターのショップブランドとしてリリースされてはいるが，外国製のマシンをそのまま日本で売り出したといった雰囲気のサブノートだ．マニュアル以外の日本語化はされておらず，キートップの標記は英語のみ．付属ユーティリティのインストールも英語モードで行うため，初めてのユーザにはとっつきにくい印象を与えるかもしれない．

本体はB5サイズで重量約1.7kg．気軽に持ち歩くにはちょっとヘビーなミドルクラスだ．しかし，ダテに重いわけではない．9.6V 2000mAhという大容量のニッカドバッテリーを内蔵しており，カタログによれば連続駆動時間は4時間にもなる．実際に，1分間隔でHDDにアクセスしながらバッテリー駆動時間を計るnenpi.exeで実験してみたところ，バックライトを点灯させたまま3時間18分動き続けた．

CPUはCyrix Cx486SLC/25MHz．試用したマシンは4MBメモリに170MBハードディスクというスペックだったが，スピード面でのストレスはほとんど感じなかった．メモリを8MBに増設するだけで，Windowsマシンとして十分に通用するパフォーマンスだ．

液晶ディスプレイは640×480，モノクロ64階調表示．反応速度は比較的速いし，上下左右とも45度前後の角度までなら見にくくなることはなかった．ただバックライトがちょっと暗めなので，明るい場所ではコントラストを調整してやる必要がある．

拡張性のポイントとなるPCMCIAカードスロットは，TYPEⅡなら2枚，TYPEⅢなら1枚セットできるタイプ．ほかにもシリアルポートや外部キーボード，それに1,024×768，16色まで表示できるCRTディスプレーインターフェースが用意されている．



▲GemPower486S/25

付属のディスクには，外部CRT用のディスプレイドライバとパワーマネージメントユーティリティ，それにPCMCIA関係のツールなどが収められている．パワーマネージメントシステムはBIOSに組み込まれているが，この付属ユーティリティを使えば，システムの起動中に設定を変更することも可能だ．

スペースキーの横にあるトラックボールと，その右側に配置されたマウスボタンは思ったより使いやすい．この位置だと，たとえば車や電車の中で膝の上に載せた状態でも，ホームポジションから手を離さずに右手親指で扱える．マニュアルには人差指で操作する図が載っているが，実際にWindowsを使ってみると，右手親指とショートカットキーの組み合わせのほうが操作性はいいようだ．

キーは比較的大きく，ストロークも深め．ラフなタイピングでも隣のキーを押してしまうようなことはない．ただ使用頻度の高いバックスラッシュキーがスペースキーの左側に追いやられているのは問題かもしれない．本来とはかけはなれた位置にあるため，慣れるにはかなり時間が必要になる．

気になるのは全体的な作りが少し頼りないという点だ．今回試用したマシンでは，BIOSやディスプレイドライバがなんとなく不安定かな，という印象を受けた．外観ではコネクタ類のカバーやPCMCIAカードのリリースレバーが，いかにもチープ．頑丈そうにできているFDDユニットも，フタが固定されていないためゴミが入りそうで不安になる．本体，FDDユニット，ACア

●GemPower 486S/25モデル170データ

CPU	Cx486SLC/25MHz
メモリ	4MB(最大20MB)
HDD	170MB
FDD	外付け標準装備
ディスプレイ	64階調モノクロ液晶，640×480
PCMCIAスロット	TYPEⅡ×2またはTYPEⅢ×1
電源	ACアダプタ，NiCd
外形寸法・重量	252(W)×191.5(D)×36(H)mm，1.7kg
その他	日本語マニュアル付属
価格	292,000円
問い合わせ先	㈱クラスター／℡03-3257-3261

●ベンチマークテスト結果

・WindSock3.30
CPU：202／VIDEO：64／DISK：92／MEMORY：78／ALL：126

・3DBench	10.8
・nenpi.exe	3:18

ダプタそれぞれのバランスは，いま一歩だろう．

マニュアルについても改善の余地がありそうだ．試用したマシンにはオリジナルの英語版をそのまま翻訳した日本語マニュアルが付属していたが，BIOSの設定やパワーマネージメントに関する記述が不足している．ただ単に翻訳するだけでなく日本のユーザのためにも，ある程度の付加情報は盛り込まれていればよいのだが，というのが正直な感想だ．

（斉藤重喜）

i486DX/33MHzを搭載した高速サブノート

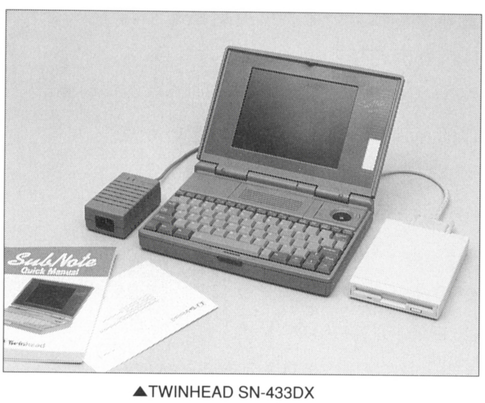
TWINHEAD SN-433DX

アサヒシステム

アサヒシステムの「SN-433DX」は，i486DX/33MHzを搭載したB5サイズ，重さ1.8kgのサブノートだ．このマシンは標準でメモリ4MB（最大20MB），内蔵ハードディスク80MB（最大170MB）を装備している．また，表示部分には8インチ液晶モノクロディスプレイを使用し，最大640×480/64階調の表示ができる．ビデオチップにはCIRRUS LOGIC社のCL-GD6412（VRAM 256KB）を使用しているので，外部モニタに接続することにより，最大800×600/16色までの表示が可能である．オプションでFAXモデムを内蔵することも可能．ほかにPCMCIA2.0（TYPE II）スロットを1基内蔵しており，PCMCIA2.0/JEIDA4.1のカードの使用も可能だ．

サブノートを開いてみると，まずすっきりとしたキーボードが目につく．このマシンでは，[Fn]キーを使用して，電源のオン/オフを行ったり，輝度の調整などを行えるが，これも見やすくキートップに書いてある．液晶ディスプレイの脇には，バッテリー残量やキー動作を示す小さい液晶のインジケータもついている．キーボード周りは全体的に好印象をもった．

キーボード右奥にはトラックボールが配置されているが，残念ながらあまり使いやすくはなかった．ボールの脇には「ドラッグボタン」と呼ばれる，1回押すとドラッグ，もう1回押すとドラッグ解除というボタンがついているが，肝心のクリックボタンがマシンのサイドについていて，クリックするたびに指を脇に動かさなければならないためだ．せっかくドラッグボタンがあるのだから，もう少しトラックボールを使う気にさせるようにしてほしいと感じた．



▲TWINHEAD SN-433DX

さて，実際のマシンの動作は，さすがに現在のサブノートでは最速のi486DX/33MHzが載っているだけあり，Windowsを動かしても速度的にはまったく問題ない．ごく普通のデスクトップのi486DX/33MHzマシンと変らないだろう．ベンチマークの結果もそのことを示している．

ただ，内蔵のモノクロ液晶ディスプレイの応答速度が低いために，マウスカーソルのポインタを見失ってしまうことが何度かあった．スペックからして，やはりWindowsマシンとしての需要がこのマシンを生んだとも思えるので，このディスプレイ部については改善を望みたい．

もう一点，今回試用したマシンには，残念ながら日本語マニュアルは付属せず，英語マニュアルが1冊入っているだけだった．内容自体は平易な英語で書かれているので，ある程度コンピュータのことを知っていれば，それほど苦労はしないとは思うが，それでもやはり日本語マニュアルの付属を望みたい．

（関 信浩）

●TWINHEAD SN-433DXスペック

CPU	i486DX/33MHz
メモリ	4MB（最大20MB）
HDD	80MB
FDD	外付け標準装備
ディスプレイ	64階調モノクロ液晶，640×480
PCMCIAスロット	TYPE II ×1
電源	ACアダプタ，NiMH
外形寸法・重量	254（W）×180（D）×40（H）mm，1.8kg
価格	258,000円
問い合わせ先	アサヒシステム㈱／℡03-3251-7691

●ベンチマークテスト結果

・WindSock3.30
CPU：254／VIDEO：120／DISK：156／MEMORY：186／ALL：182

・3DBench	20.8
・nenpi.exe	1:53

頑丈な作りと
高機能のディスプレイドライバが魅力

ZEOS Contenda

ZEOS

ZEOS ContendaはアメリカのZEOS社が販売するサブノートだ．ZEOSはGateway2000やDELLと並ぶアメリカ最大の通販を主体にしたAT互換機メーカーである．ZEOS社は以前にもZEOS Pocketというハンディマシンを販売していたが，ContendaはWindows環境にも対応した本格的なサブノートだ．

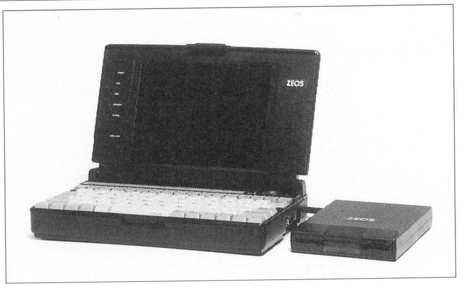
Contendaには，メモリ/HDD容量の違いなどにより三つのPACKAGEが用意されており，それぞれにCPUがi386SL/486SLのタイプがあるが，ここではPACKAGE3の486SLタイプについてみてみよう．

CPUにはi486SL/20MHzを搭載し，メモリは標準で8MB(PACKAGE1/2は4MB)，ハードディスクは80MBである．3.5インチ外付けFDDは添付されているが，PCMCIAスロットは装備していないのが残念な点だ．最も装備の多いPACKAGE3では，FAXモデム(データ部分2,400bps，FAX部分9,600bps)や予備バッテリー，キャリングケースなども含まれている．

サイズはA5よりやや大きめで，厚さはThinkPad220の約1.5倍近くある．小脇に抱えてもさほど違和感はないが，コンパクトなボディとは裏腹に重量が約1.8kgあるため，「意外に重い」という印象を受ける．黒塗りのボディは国産サブノートにないような頑丈な作りで，落としても壊れそうな気がしない(実際には壊れるだろうが)．

本体背面には，バッテリーを装着するためのスロットのほか，外部VGAポート，パラレルポート，外部FDDポート，ACアダプタのコネクタがあり，本体左側面にはシリアルポートとFAXモデムのモジュラコネクタが配置されている．

Contendaのビデオチップには Chips&Technologies社の65520が使われているが，付属のディスプレイドライバの高機能さは，ほかのサブノートにないものだ．通常，液晶ディスプレイでは640×480ドットしか表示できないが，Contendaではディスプレイドライバを変更することにより，800×600ドット/256色などの仮想スクリーンモードが使える．仮想スクリーンモードはマウスカーソルを画面が端へ行くとスクロールするもので，液晶ディスプレイ上でも高解像度の画面が使える．このほかにも，外部出力を使った1,024×768ドット/16色，Windowsアクセラレータ機能を使った640×480ドット/256色モードなどが用意されている．液晶ディスプレイには，64階調表示の可能な7.4インチサイズのものを使っている．反転表示も可能で，視認性も悪くない．外部モニタへの表示切り替えも動作中に可能だ．



▲ZEOS Contenda

本体のキーボードは，US101キーをベースにしたもので，ノートパソコンなどと同じ標準サイズだ．文字を連続して入力するときにも不満を感じない．Windows環境ではキーボード上部にあるトラックボールを使う．スイッチ類はトラックボールの左右に配置され，実用的なレイアウトとなっている．

98NOTEにあるようなレジューム機能はないが，メモリ情報をハードディスクの未使用領域に書き出すサスペンド機能があるため，途中で停止してもバッテリーの消費はない．

ベンチマークテストの結果を見ると，バッテリー駆動時間の長さとビデオ回路の速さが目につく．これだけ動作すれば，あまり不満はないだろう．PCMCIAスロットがないことを除けば，ほぼ及第点のつけられるサブノートといえるだろう．

(法林岳之)

●ZEOS Contenda(PACKAGE3 486SL)データ

CPU	i486SL/25MHz
メモリ	8MB
HDD	80MB
FDD	外付け標準装備
ディスプレイ	64階調モノクロ液晶，640×480
PCMCIAスロット	ー
電源	ACアダプタ，NiMH
外形寸法・重量	246(W)×155(D)×46(H)mm，1.7kg
価格	298,000円
問い合わせ先	くまくま／℡0975-46-3720

●ベンチマークテスト結果

・WindSock3.30(Linear Acceleration)
CPU：165／VIDEO：206／DISK：143／MEMORY：170／ALL：169

・3DBench	16.9
・nenpi.exe	02:59

FAXモデムも内蔵した
99,800円のサブノート

Massif

オムロン

モデムメーカーとして知られるオムロンのMassifは,「パーソナルインテリジェントコミュニケータ」と位置づけされているサブノートである.本体サイズは,本誌をおよそ3冊重ねてひとまわり小さくした大きさだが,これで880gに抑えるために,本体のプラスチックも特に上蓋が薄く,強度に不安が残る.

CPUは,Chips&Technologies F8680で,性能的には80286相当品である.メインメモリは疑似SRAMで1MB用意されており,増設はできない.また1MBのリチウム電池でバックアップされたRAMディスクも用意されており,JEIDA4.0対応のメモリカードスロットを使って拡張することも可能だ.コネクタ類は,モジュラジャックとRS-232Cコネクタ,それにプリンタコネクタが用意されており,モジュラジャックに電話線を繋げば,すぐにパソコン通信とFAXの送受信ができる.RS-232Cは他のパソコンとのデータ交換用に用意されているが,別売のケーブル(PC-9800シリーズ,DOS/V用ソフト付き)を購入しなければならない.プリンタコネクタは市販の98NOTE用のプリンタケーブルがそのまま使える.

OSもDR DOS5.0から必要最小限のファイルのみを残したものだ.したがってDOS環境でパソコンとして使うにはかなり不便を強いられることになるが,もっとも最初からアプリケーションをインストールして使うといったパソコンとしての用途は想定していないため,DOS環境での使用方法については一切触れられていない.

気になるバッテリー駆動時間は,単3アルカリ電池4本を利用し,最大8時間の使用できる.

●ちょっと賢いワープロ

Massifはワープロを中心に,電子手帳(カレンダー,スケジュール管理,備忘録,メモ,電卓,世界時計),モデムを1台に凝縮したサブノートだ.これらのアプリケーションは,グラフィックベースのメニューソフト"J.STAFF"から呼び出して使うように設計されており,終了後は再びメニューに戻る仕掛けになっているが,直接他のアプリケーションに切り換えることも可能だ.カレンダーと電卓,世界時計についてはパソコン通信とFAXを除く各アプリケーションからデスクアクセサリとして呼び出して参照することができるが,ワープロ使用時にはカレンダーから世界時計まで電子手帳機能をすべて呼び出して参照することができる.しかし,呼び出すことはできても参照だけで,そこにあるデータをコピーしてワープロにペーストすることはできない.こういうことができれば,かなり便利ではないかと思う.



▲Massif　他のサブノートとは一線を画す

ワープロには286環境でも快適に使えるということでVJE-Penが採用されており,日本語入力に使う仮名漢字変換ソフトにも同様の理由でVJE-Σが採用されている.

●パソコン通信とFAXの実力は

ところで内蔵のFAXモデムの実力はどんなものなのだろう.モデムメーカーの製品だけに,気になるところだ.このFAXモデムはデータモデム部分は2,400bpsの通信速度で,MNP5またはV.42bisのデータ圧縮機能を持つ.FAX部分は送信9,600bps,受信4,800bpsでG3をサポートしている.現段階では日常使用に十分な性能だろう.

通信ソフトは,四大商用BBSの電話番号が登録されており,オートログイン機能が使える.残念ながら新規登録したBBSではこの機能は使えない.

FAXについては,カバーページの設定も用意されており,ワープロから直接送信できることもあってなかなか使いやすいが,受信したFAXは液晶に表示すると

●Massifデータ

・CPU	F8680/8MHz
・メモリ	1MB
・HDD	—
・FDD	—
・ディスプレイ	反射型2階調モノクロ液晶,640×400
・PCMCIAスロット	TYPE II ×2(JEIDA4.0)
・電源	ACアダプタ,単3電池×4
・外形寸法・重量	235(W)×178(D)×35(H)mm,880g
・その他	VJE-Σ,VJE-Pen,電子手帳,FAXモデム装備
・価格	99,800円
・問い合わせ先	オムロン(株)情報機器事業部/TEL03-5488-3221

●ベンチマークテスト結果
・測定せず

き等倍で表示した場合，1ページを分割表示することになり，表示の切替が遅く，かなりストレスが溜る．

全体的にみて，初心者にも難しすぎず，機能もまとまったよくできたサブノートだ． （戊野速太朗）

拡張性とオプションに期待できるB5ノート

AV-B5 NT

キャラベルデータシステム

98用ハードディスクなどの周辺機器メーカーとしても知られるキャラベルのサブノートが『AV-B5NT』だ．AV-B5NTはその名の通りB5サイズのサブノートである．マシンそのものはエスエスケーベルから発売されているSWIFTと同じであり，拡張性に優れている点，一部のキーの押しにくさといった点についてはSWIFTの項で触れられている通りだ．重複するため，ここでは細かいレビューは割愛する（外形寸法の公表値が若干違うが，測定時の誤差だろう）．なお，今回試用したマシンは，プロトタイプのため，ベンチマークについても割愛する． （法林岳之）

●AV-B5 NT120Wスペック

・CPU	i486SX/25MHz
・メモリ	4MB（最大20MB）
・HDD	120MB（着脱式）
・FDD	外付け標準装備
・ディスプレイ	16階調モノクロ液晶，640×480
・PCMCIAスロット	TYPE II ×1
・電源	ACアダプタ，NiCd
・外形寸法・重量	280（W）×177（D）×37（H）mm，1.8kg
・価格	248,000円
・その他	DOS5.0，Windows3.1（ともに日本語版）プリインストール
・問い合わせ先	㈱キャラベルデータシステム／℡03-5561-9361

●ベンチマークテスト結果
・測定せず

Win/Vの登場が待たれるスーパーサブノート

OmniBook300

横河・ヒューレット・パッカード

日本アイ・ビー・エムのThinkPad 220が日本を代表するサブノートとすれば，このヒューレット・パッカード（HP）のOmniBookはアメリカ，いや世界を代表するサブノートといっても過言ではない．

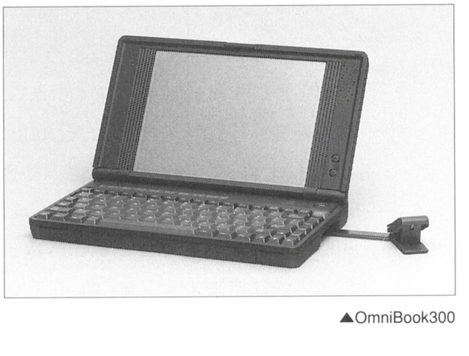
HPにはご存じのようにHP 95LX/100LXというパームトップPCがあるが，そのノウハウを活かし，100LX約3個分の大きさにまとめたマシンがOmniBook300だ．HP 100LXの3個分だから300というネーミングになったという話もあるが，実際の寸法はB5サイズに近い．重量は約1.3kgとThinkPad 220より重いが，手に取るとそれを感じさせないバランス感がある．

OmniBook300はCPUに386SXLV/20MHzを採用し，標準で2MBのメモリを搭載している．記憶メディアは，10MBのフラッシュディスクを内蔵したモデルと40MBのハードディスク（PCMCIA2.0 TYPE III）を内蔵したモデルの二つのバリエーションがある．増設FDDはオプションにもなく，外部とのファイルのやり取りはPCMCIAカードか，後述する“LapLink Remoto Access”を使う．

液晶ディスプレイは反射型の16階調のもので，VGA相当の表示が可能だ．反射型といえ，視認性はかなりよく，安物のカラー液晶よりも目が疲れない．バックライトがないため，バッテリー駆動時間も他のサブノートより長い．内蔵のNiMH充電池パックで，ハードディスクモデルなら6時間，フラッシュディスクモデルなら10時間まで駆動することができる．代用電源として単3リチウム電池（あるいはアルカリ電池）4本で駆動することも可能だ．

●I/O，拡張性にも優れたマシン

本体にはPCMCIA2.0 TYPE II スロット×4（またはTYPE III ×2）があり，ハードディスクモデルではTYPE II ×1，フラッシュディスクモデルではTYPE II ×2を空きスロットとして使用できる（TYPE II スロット×1はSYSTEM ROMが使用）．本体後部には，D-Sub9ピンのシリアルポート，D-Sub25ピンのパラレルポートのほかに，HP 95LX/100LX，



▲OmniBook300

同社のデスクトップ機Vectraシリーズと通信するための双方向赤外線ポートを装備している．

本体のキーボードはデスクトップ機などで使われているものとほぼ同じサイズで，連続してキータイプすることも苦にならない．また，Fn キーとF1～F9キーを組み合わせることで後述するROM版のアプリケーションを起動することができる．

OmniBook300には他のサブノートにない特徴がいくつかある．第一の特徴は，ソフトウェアをROMで持っている点だ．すべて英語版だが，MS-DOS5.0，Windows3.1，Excel4.0，Word2.0などをROMで搭載している．そのほかにも，Appointment Book，HP Financial Calculatorなどもあり，ビジネスマンが仕事に使いそうなソフトがすべて含まれている．英語版であるため，日本語を使うことはできないが，C.F.Computingが発売を予定しているWin/Vを使うことで，日本語版Windowsアプリケーションソフトを使うこともできる．Win/Vは英語版Windows上で日本語版Windows用ソフトを使えるようにするもので(本誌「DOS/V Paradise」などを参照)，'93年12月よりβ版テストが開始されている．

●使いやすいポップアップマウス

第二の特徴はやはりポインティングデバイスだろう．サブノートのポインティングデバイスというと，トラックボールが主流だが，OmniBook300にはポップアップマウスというものがついている．このマウスは，キーボード上部にあるネズミのマークのボタンを押すと，ポンと外へ飛び出す．本体とは薄い板のようなものでつながっており，普通のマウスと同じように操作することができる．小さいので操作しにくいのではないかと思われがちだが，実は意外に操作しやすく，ヘタなトラックボールより快適に使うことができる秀逸なポインティングデバイスなのだ．

ところで，外部とのデータのやり取りだが，ICカードを媒体にする以外にもいくつかの方法がある．まず，上述のように同社のHP 95LX/100LX，Vectraシリーズとは後部の赤外線ポートを使って通信をすることが可能だ．結線が必要ないのだから，こんなに便利なことはない．アメリカではこの赤外線ポートを他のデスクトップ機に増設する“Jet Eye”といった周辺機器も販売されているようだ．

もうひとつの方法がLapLink Remote Accessを使ったリモートドライブという方法だ．日本でいうところのMAXLINK(メガソフト)などと同じような使い方をする．しかし，ソフトウェアはOmniBook300側からインストールすることができるので，本体とシリアルケーブルさえ持っていれば，どのパソコンともやり取りが出来てしまうわけだ．ソフトウェアはWindowsの中からインストールするのだが，指示に従って操作すれば，簡単そのもの．速度も最大115,200bpsと速いため，逆にフロッピーディスクを使う環境よりも快適に感じることもあるようだ．

さて，実際に操作をしてみると，DOS環境では不満がないものの，Windows環境を使うにはややツラいという印象は否めない．ベンチマークテストの結果を見ても分かるように，Windows環境では全体的に今ひとつキビキビ感が足らないのだ．ソフトをROM化してるとは言え，Windowsを使うにはCPUパワーが不足していると考えるべきだろう．

ところが，本稿執筆中に，アメリカからOmniBook300の486版“OmniBook 425”が発売されたというニュースが飛び込んできた．基本的な構成はOmniBook300と同じだが，CPUがTI社の486SXLVに変更されている．価格もハードディスクモデルが2,125ドル，フラッシュディスクモデルが2,375ドルとなっている．今後，486版OmniBookが日本国内で販売されるかどうかは未定だが，サブノートファンの期待に応えるためにもぜひ発売してほしいところだ．もちろん，Win/Vもバンドリングされていれば，申し分ないだろう．

(法林岳之)

●OmniBook300(40MB HDDタイプ)データ

・CPU	AMD386SXLV/20MHz
・メモリ	2MB(最大8B)
・HDD	40MB(＊)
・FDD	—
・ディスプレイ	16階調モノクロ液晶，640×480
・PCMCIAスロット	TYPEⅡ×1
・電源	ACアダプタ，NiMH
・外形寸法・重量	282(W)×163(D)×36(H)mm，1.36kg
・その他	DOS5.0，Windows3.1，Word2.0，Excel4.0，LapLink Remote AccessなどをROM組み込み済み(すべて英語版)
・価格	310,000円
・問い合わせ先	横河・ヒューレット・パッカード㈱／℡03-3335-8333

●ベンチマーク結果

・Windsock3.30
CPU：77／VIDEO：54／DISK：29／MEMORY：103／ALL：65

・3DBench	6.2
・nenpi.exe	測定せず

＊10MBフラッシュディスクモデルもあり．価格は380,000円．

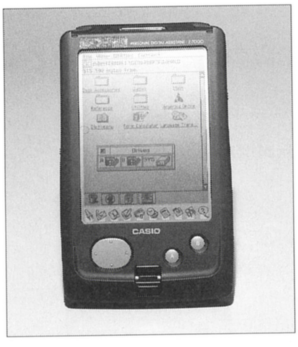
以上，11のマシンについてみてきたが，時間的な制約などでレビューできなかった機種がほかにもたくさんある．ここでは，速報として次の3機種を紹介する（編集部）

米CASIOのペン入力タイプPDA

ZOOMER（Z-7000）

■米CASIO社が販売する「Zoomer（Z-7000）」は，OSにGeoWork社のGEOS Ver.2.0を採用した，ペン入力タイプのPDA（Personal Digital Assistant）だ．

Zoomerは，CPUに8086相当品を使用し，メインメモリ1MB，PCMCIA2.0 TYPEⅡスロット×1などが搭載されている．またAmerica Onlineに接続するための専用ソフトをはじめ，住所録や翻訳ソフトなどのアプリケーションが，ROM上にインストールされている．



このマシンの特徴のひとつが，最近，電子手帳やOmniBook赤外線送受信端子を内蔵していること．9,600bpsでZoomer同士を接続することが可能である．

気になるバッテリーでの駆動時間であるが，単3乾電池3本で約100時間と，使っている間にバッテリーの心配をしなくてもいいレベルである．またどこでも購入できる乾電池というのもうれしい．

サイズはB6よりわずかに小さい176.2（W）×107.6（D）×26.0（H）mm，重さは430gで，アップルのNewton MessagePadとほぼ同じである．赤外線送受信端子といい，かなりこのNewtonを意識したものであることは間違いなさそうだ．価格は895ドル

GATEWAY2000社のサブノートが日本上陸

日商岩井がHANDBOOK486を発売開始

■日本でも有名なAT互換機メーカーGATEWAY2000社のサブノートを日本でも手に入れることができるようになった．日商岩井が12月1日より，「HANDBOOK 486」を発売する．



このたび発売されるマシンは，CPUにSLエンハンスi486SX/25MHzを使用，メインメモリ4MB（最大20MB），HDD80MB，PCMCIA2.0 TYPEⅡスロット×1を搭載する．HANDBOOK486には486DX2/40MHzを搭載したモデルもあるが，こちらの発売時期は未定．

サイズは248×150×40.6mm，重さは1.3kg，ディスプレイは640×480，FSTN白黒液晶である．

英語版MS-DOS6.0/Windows3.1が標準で付属する．価格は248,000円

（問い合わせ先：日商岩井㈱PCグループ／℡03-3588-2419）

東芝アメリカ情報システム社がサブノートを発売

日本での発売はいつ!? T3400シリーズ

■東芝アメリカ情報システム社が，サブノート「T3400」2機種（カラー/モノクロ）を発売した．



サブノートとして初めてTFTカラーディスプレイやローカルバスグラフィックアクセラレータ（WDのRocketCHIP WD）を搭載するなど，意欲的なマシンだ．その他の特徴としては，CPUにSLエンハンスi486SX/33MHzを搭載，ポインティングデバイスに，IBM ThinkPad750Cなどで採用されているスティックタイプのものを採用していること，バッテリーにリチウム2次イオン電池を採用し，駆動時間の延長を実現していること，などがある．PCMCIA2.01 TYPEⅡスロットも装備．

サイズは252×201×461mm，重さは約2kg（カラーモデル）．価格はモノクロモデルが2,599ドル，カラーモデルが3,999ドルとなっている．

COMDEX / Fall '93 PCMCIAプロダクツレポート

●著者●
佐々木 哲
Satoru SASAKI

●概要●
ノートブック，サブノート，PDAとハードウェアの進歩により小型高性能化が進んだ．
しかし，使い方の進歩なしには何の意味も持たない．
これら携帯機器の“ATバス”として規格されたPCMCIAは，すでに標準インターフェイスとして定着した．
そして，今やその製品はメモリに留まらず各種I/Oカード百花繚乱の時期を迎えた．

PCMCIA製品の多くは，そのまま日本のDOS/V互換機ですぐ使えるものです．中には，モービルコンピューティング，無線利用の製品のように，日米のインフラ整備の差に愕然とさせられるものもあります．また，I/Oカードが普及するにしたがって，その互換性が問題になってきましたが，その解決方法も検討されています．そこで，本稿では先月開催されたCOMDEX/Fall '93に出展された新製品を中心に，PCMCIA製品の動向やなどについて見ていきましょう．

まず，I/Oカードを中心に新製品を見ていきましょう．とにかく，基本的に今までデスクトップであったカード類は，需要さえあれば全てPCMCIAになると考えて間違いありません．ここでも，モデム，SCSI，サウンド，LAN，ハードディスクと何でも揃っています．価格的にも，もうPCMCIA I/Oカードは高価なものではありません．

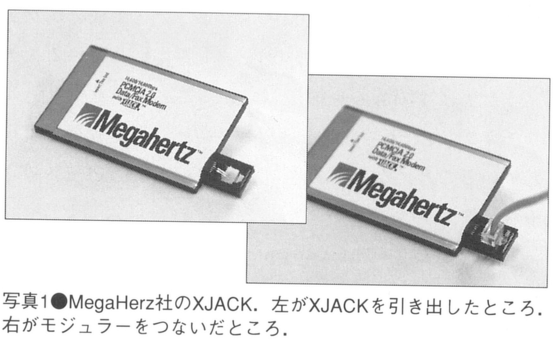


写真1●MegaHerz社のXJACK．左がXJACKを引き出したところ．右がモジュラーをつないだところ．

●メモリカード

容量の増加もあまりなく，安定しているようです．アプリケーションの配布に向いているOTPメモリ(One Time Programmable：一回だけ書き込み可)もそろっています．SRAM，DRAM，フラッシュメモリ，OTPを一枚に自由に組み合わせられるものもあります．

●FAXモデム

デザイン面でMegaHerz社の「*XJACK*」(**写真1**)にかなうものはないようです．つまり尻尾のないモデムは今の所これだけです．PCMCIAモデム本体は，従来のモデムのようにV.32bis/V.17 (DATA/FAX 14,400bps)は当たり前で，V.32terbo(19,600bps)やV.fast(28,800bps)製品が発売されるのも時間の問題でしょう．

日本との大きな違いは，携帯電話との接続性です．携帯電話に直結できるFAXモデムが数多く紹介されました．これは日米の携帯電話環境の大きな違いの一つで，後をひきそうです．日本のものは残念ながらとても接続しにくくなっています．

XJACKに対抗するものとしては，カードからいきなりRJ-11ジャック付きの電話線が出ているものや，コンパクトなRJ-11ジャック(Line, Phone)を組み込んだものがありました．

また，DSPを使ってサウンドカードとFAXモデムを統合するのが流行っています．つまり一枚のカードでモデム，FAX，音声入出力，サウンドカードの機能を

持てるのです．CIM Engineering社の「*CFV-9624C*」は，DATA 2,400bps/FAX 9,600bpsのFAXモデムとしての機能に加え，電話線より音声を録音再生できます．つまり，これ一つでノートブックをホストとした留守番電話にもなれば，音声応答によるFAX引出システムも作れます．音声は拡張ATコマンドでコントロールするようです．

●ISDN

ドイツのモデムおよびPCMCIAスペシャリスト，Dr. Neuhaus社からはISDNカード「*NICCY CARD ISDN*」が出展されていました．2Bチャンネルでいきなり128kbpsの通信の世界です．同社からは，さらにDigital Cellular phoneであるGSM対応のPCMCIAモデムも発表されています．

●LAN

PCMCIAイーサネットカードはもう豊富に出まわっています．10Base2，10Base-T，トークンリングにいたるまで選べます．とりあえず，NE2000互換といっているものとしては，Accton Technology社の「EtherDuo-PCMCIA」があります．

また，PCMCIAスロットが一つしかないのに，モデムも使いたいし，LANにもつなぎたいという人のためのカードがありました．Socket社の「*Socket TRIO Credit Card Adapter*」(**写真2**)は，14.400bpsのFAX/DATAモデムとイーサネット10Base2/TのLANアダプタを一つにしたカードです．

また，Xircom社より「*Netwave*」という無線LANが出ました(**写真3**)．2.4GHz帯を使い，PCMCIAカードにコネクタサイズの小さなアンテナがついた製品です．

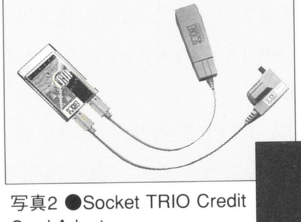


写真2 ●Socket TRIO Credit Card Adapter



写真3 ●Xitcom Netwave

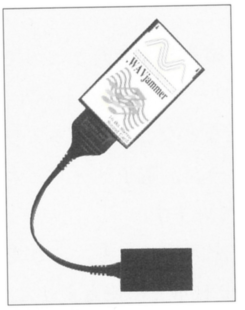


写真4 ●.WAVjammer



写真5●SCSI2GO

これで面倒な配線無しで場合によっては室内を移動しながらでもLANにつながります．同製品は「*Netwave Access Point*」という親機で，20台ほど子機を管理でき，親機を複数使う場合のRoamingまでサポートしています．親機を既存のイーサネットあるいはトークンリングLANに直接接続します．室内で3～50メートル飛び，速度は1Mbps．NetWare，LAN Managerサポート．PCMCIAスロットのない機種のため，パラレルポート接続版があります．

●サウンド

ノートブックやサブノートでもマルチメディアやゲームはします．PCMCIAサウンドカードもすでにいくつもありますが，SoundBlaster互換のものを一つ紹介しましょう．New Media社は，SoundBlaster，MS Sound System，Adolib 互換のサウンドカード「*.WAVjammer*」(**写真4**)を出展．コンパクトなケーブルの先にステレオ，ラインIN/OUT，マイク，ヘッドフォンジャック付き．44.1KHz 16ビットステレオサンプリングが可能です．これでポータブルなゲーム，プレゼンテーション環境ができ上がります．

●SCSI

New Media社の「*Visual Media Card*」や日本のマクニカのように低速のSCSIカードはすでに発表されています．今回はFAST SCSI-2サポートのものが4種類ほどあり，唖然としました．CD-ROMなど低速なものに限らず，ハードディスクをメインストレッジとしても十分使えますし，スキャナなどからの入力も可能です．その中でFutueDomain社は，DOS，Windows，OS/2に対応した「*SCSI2GO*」(**写真5**)を出展していました．

●モービルコンピューティング

この分野が最も日米の差が明確化してきたところです．すでに全米中で独自ネットワーク，ポケットベル，双方向Paging service，携帯電話などなど各種ネットワークが張り巡らせられ，メッセージ，電子メール，ニュース，データ等情報配信されています．

いくら端末側のハードウェアを入手しても，これだけのインフラが整備されない限り日本では使えません．残念ながら，日本国内に環境が整備されるには長い時間がかかりそうです．とても危機感をおぼえます．

Motorola社などは，これらのサービス全てに対応できるよう，各種デバイスを整えています(**写真6,7**)．たとえば「*Embarc*」と呼ばれる単方向データ転送システムのPCMCIAレシーバは249ドルで1,200bpsでデータ受信できます．すでにアメリカ国内では，特定ユーザーへのメール，データ転送に加え，ニュース配信放送も始まっています．たとえば「USA Today」やロイターのニュースなどが配信されています．受信機器はこれを取り込みユーザーは好きなときに好きな部分を見るだけです．このサービスに加入するには，アダプタを購入し，月々18ドルの基本料を支払うだけです．

双方向Pagerもシステム整備が進んでいます．これでポケベルの一方通行でなく，センターを介して移動端末同士で連絡できるのです．またARDISのように全米に無線網を張り巡らせた双方向サービスもあります．

新しいところでは，既存のアナログ携帯電話網を利用し，19,200bpsの速度でデータをパケット通信できるデジタルパケットデータシステムCDPDも規格が決まり，整備が進んでいます．これは専用のモデムを既存の携帯電話に接続して使います．素晴らしいのは，信頼性の高いパケットデータ転送を電波の空きを利用して安価に行うことです．すでに携帯電話の周波数帯の不足が報じられるアメリカですが，それでも無音中など最低30%の時間は電波があいています．CDPDはこの隙間を利用してデータパケットを飛ばすため，使用料金は時間ではなくパケット単位で課金され，非常に経済的です．



写真6●NewsCard&NewsStream

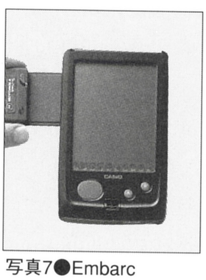


写真7●Embarc

これらのインフラを使ったサービスも急速にあらわれてきています．先に述べたニュース配信などのほかにも，radiomailのように月に数10ドルの契約料でInternetをはじめ，どの電子メールとメールのやりとりが無制限にできるものなどがあります．無線の特徴として，複数の相手に対して同一内容のメールを送る場合，一度の送信ですむため，使い方によっては有線の電話やFAXよりも格安になる場合もあります．

携帯電話が200ドル台で買え，通常の電話料金の範囲で活用できるアメリカだけに，これだけの応用・発展が進んでいます．

たとえば，Socket Communications社の「*Socket PageCard*」は，PCMCIAインターフェース付きのポケベル．単体でも24文字表示LCD付きの普通のPagerとして使え，更にPCMCIAスロットにさした状態ではアプリケーションからメッセージやデータを読み込みます．さらに一方通行のODI, NDISドライバも用意されています．つまり，NetWareでも使えるわけです．また，DOS, Windows, GEOSもサポートされています．

●PCMCIA ATA

ATAとはIDEドライブのようなデバイスインターフェースのANSI規格です．PCMCIAでもATAの規格があり，ハードディスクの他に，Sundisk社のフラッシュメモリディスクがあります．これらは当然のことながらPC側からはI/Oを通してIDEドライブとしてみえます．ですから，当然PC側がPCMCIAでATAデバイスをサポートしていないと使えません．

PCMCIA HDはPCMCIA TYPEⅢのサイズの中に，IDEハードディスクを詰め込んだものです．現在発売されているものの中では105MBくらいが最大ですが，すでに170MBモデルを発表した所もあります．Sundisk社も新しく40MBのFlash Disk「*SDP5A*」を発表しました．

これらのデバイスを生かすためには，PCの電源が入っている時に抜き差しできるHot swap機能がシステム

側でサポートされる必要があります.

とんでもないATAデバイスとしては，SyQuestのPCMCIA版がありました．リムーバブルハードディスクで長い実績を持つSyQuest社ですが，なんと新たに1.8inchのカートリッジを作り，TYPEⅢサイズに収めました．カードそのものをPCMCIAカードとしてPC本体から抜き差しできますし，取り出したカードからHDカートリッジを取り外たり交換することができるのです．容量は80MBと60MBがありました.

●GPS

GPSカードもいくつか出てきています．そう，あの衛星からの位置情報を得るシステムですね．これもPCMCIAカードになります．モービルコンピューティングに最適な必須デバイスともいえましょう.

Socket社の「*Mobil GPS*」は2メートルのケーブルに直径6cmの外部アンテナ，サンプルGPSアプリケーション(DOS，Windows)が付属(**写真8**)．SDKオプション(C library. DOS, Windows, PenPoint)も用意されています．DOS, Windows, GEOS, UNIX, OS/2, SCO, Solaris, PenPointと，幅広いOSに対応しているのも魅力です.

●GPIB

研究室など機器コントロール，データギャザリングには欠かせない，GPIBインターフェイスもあります．コンパクトでEMIの少ないNotebookは最適のコントローラになるでしょう．National Instruments社の「*PCMCIA GPIB*」は，TYPEⅡのPCMCIAカードから少し太目のケーブルが伸び，その先に例のごついGPIBコネクタがつきます．同社の「*LabView*」から使えるほか，開発ライブラリもあります.

●外部接続デバイス

これだけ各種PCMCIAカードが出てくると，データ交換のためにデスクトップPCに接続できるととても便利です．このための‘カードドライブ’というべき製品が多数出てきています．多くはデスクトップPCのドライブベイにつけるユニットと，ISAバス用のインターフェイスカードからなります．また，外付けのものやパラレルポート接続のものもあります.

そのドライブがメモリカードだけか，あるいはI/O，ATAカードまでサポートするのか気を付けないといけません．内蔵のものはメモリ，フラッシュメモリ，I/O，ATAカードと全てサポートします．パラレルポート接続のものは原理上I/Oカード，ATAカードは使えません.

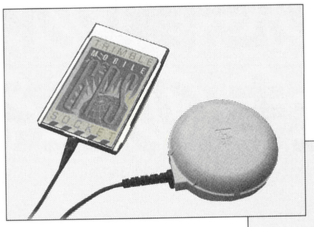


写真8●Mobil GPS

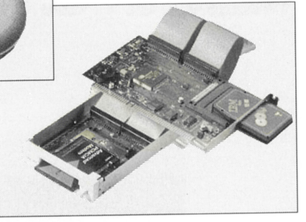


写真9●Cardport isa

I/O，ATAカードがサポートされるものは，PCMCIA LAN，モデム，SCSIなどすべてそのまま使えますし，Hot Swapも可能です．PCMCIAカードが何枚使えるかも気になります．TYPEⅢながら，TYPEⅠでも一枚しか使えないものや，逆にTYPEⅢより少し厚めにしてTYPEⅢとTYPEⅠ,Ⅱが同時に使えるようにしたものなどもあります．このようなドライブは，すでに300ドル以下で出まわりはじめています．その中のいくつかを紹介しましょう.

Data I/O社は，パラレルポート接続の「*CardPro 1*」(TYPEⅡ)．3.5inch FDサイズのユニットにTYPEⅢ(オーバーサイズ)まですべてのPCMCIAカード2枚サポートした内蔵型の「*CardPro 2*」とそのデスクトップ版を出展していました.

Chase advanced technologies社の「*CARDport*」も内蔵，外付け2種類あります．それぞれ，DOS，Windows，OS/2をサポート．「*CARDport eX*」はパラレルポート接続で，すべてのメモリカードをサポート.「*CARDport isa*」は，すべてのTYPEⅢまでのカードをサポート．加えてISAインターフェイスカードにも後ろから1枚ささる内蔵用ユニットです(**写真9**).

このほかにも，Adtron社がOEM向け含め，パラレル，シリアルポートにつながる外部式のもの．IDEインターフェイスにつながりメモリデバイスをサポートするものを発表．システムソフトウェアやチップを供給するDatabook社も内蔵，外付けモデル，ISAカードのみのものと各種カードドライブを出展していました.

中村正三郎のこの人に聞きたい❶

自分の想像の限界を超えたものこそ面白い

数学に魅せられた女(ひと) 木本圭子氏



木本圭子さんとは，あるパーティで知り合った．ハッカー会議*[1]議長のグレン・テンニーさんが来日し，彼を囲む会が新宿で催された，そのときに知り合ったのだった．

たまたま隣になり，話をはじめた．

中村「ご職業は」

木本「デザイナーです．雑誌のレイアウトなんかもやってます」

中村「パソコンはお使いですか」

木本「ええ，Macを使ってます」

こんな具合にはじまった．

ある日突然目覚めちゃった女(ひと)

職業はデザイナー，パソコンはMac，となれば容易に連想できるではないか．PhotoShop，Illustrator，QuarkXPressを使ってお仕事といった，よくあるギョーカイさんだ．

PhotoShopで写真をきれいに仕上げ，Illustratorで絵をちょいちょいのちょいで描き，DTPソフトのQuarkXPressで出版物のレイアウトを決めていく．ギョーカイさんがMacを使っているといえば，まずこれだ

これには，ぼくはいささか食傷気味だ．話はできるが，興味をもって話すというより，何も話さないのは失礼だと思うから話すといったものになりがちだ．期待せずに，木本さんとの会話も無難なところから攻めてみることにした．

中村「じゃあ，PhotoShopとか使って，ごにょごにょと」

木本「うーん，まあ，使いますけれど，私がMacを熱心に使うのは，別のことで」

中村「別のこと?」

木本「コンピュータグラフィック(CG)っていうんで



*1 世界中からハッカー（よい意味でのハッカー）が集まり，三日三晩語り合い，情報交換をし，親睦を深めるイベント．'94年は10周年記念大会になる予定．

*2 ここまでひどいと数学が弱いとはいわず，算数が弱いという．

*3 マイクロソフト社とのトラブルから，連載中止になった私の連載．あのパーティのころは，まさか連載中止になるとは思っていなかった．

*4 中村，古いね．チッチとサリーについては，誰か，女の子に聞こう．昔，女の子だった人もきっと詳しいぞ．

すか」

ぼくは，またかと思う．MacでCG．これもパターンだ．いるんだよ．お絵かきソフトでちょこっとやっただけで，CGやってますって顔してるのが．

中村「あ，じゃあ，Illustratorとかで?」

木本「いえ，THINK Cでやってます」

中村「えぇっ!!」

しんく・しぃっ!?

のけぞった．

THINK Cといえば，シロートさんが使うMac用ソフトを作るために，プロが開発に使うものだ．Cというプログラミング言語のコンパイラというものと，開発環境が一緒になっている代物ですよ．なんなんだ，この女(ひと)は．

中村「THINK Cですかぁ」

木本「ええ，Cでプログラムを書いて，グラフィックをやるんです．たまたま，今日，いくつかもってきているので，お見せしましょうか」

中村「ぜひぜひぜひ」　いくつかの作品を見せてもらった．美しいと思った．

中村「これ，どうやって計算してるんですか」

木本「えっと，それは，やっぱり方程式があって，それにパラメータを」

方程式．ああ，なつかしい響きだ．

プログラマは一般に数学に強いと思われているが，少なくともぼくは弱い．プログラミングの上では，足し算引算がほとんどで，それも1足すか1引くことが多い．割算なんて滅多にやらない．方程式云々は，とうの昔に退化してしまっており，実生活では2桁足す2桁が，もうむずかしい．2桁引く2桁になると答えがしばらく出てこない．*2

対照的にこの女(ひと)は，おれ様の苦手分野に挑戦し，Macを駆ってグラフィックにしているというのか．

中村「へぇ，それじゃ，理系の学校だったとか，昔から，数学は得意だったとかって」

木本「ぜーんぜん．数学なんかぜーんぜん」

中村「えっ，だったら，ある日，突然?」

木本「ある日，突然」

中村「目覚めちゃった」

木本「目覚めちゃった」

中村「数学に」

木本「はい」

中村「独学で」

木本「はい」

なのだそうである．

この時点で，「電脳曼陀羅」*3で紹介してみたいと思った．グラフィック云々もさることながら，数学をまるで知らなかった女(ひと)が，グラフィックで自分を表現したいために，目覚めていく．そこに興味をひかれた．

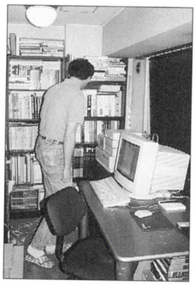
後日，レインボーブリッジの夜景も見える木本さんのお宅にお邪魔して，あらためてお話を伺った．

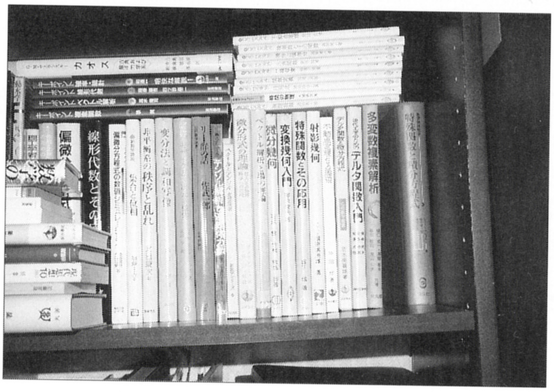
MacPaintに限界を感じてプログラミングを始める

まず本棚が尋常ではない．飛んだり跳ねたりしているわけではない．中身だ．

ほとんど積ん読だと木本さんは謙遜するが，積ん読にしても，解析学の入門書から，偏微分方程式，テンソル，量子力学やらあれやこれや，この種の本が本棚を占拠している．明らかに偏見だが，到底，女性の家の本棚とは思えない．チッチとサリーくらいおいておけよ．*4

数学など，まるで知らなかったし，知ろうとも思わ





(写真左)システムのチェックをする前に、いきなり本棚のチェックを始めてしまった筆者(笑).

(写真右)いちから始めてこれだけ徹底するのは、やはりすごいものがある．このほかに、『多様体の基礎』『微分幾何入門』(ともに東京大学出版会),『解析概論』(岩波書店),『関数論』(ジョージ・アルフケン著，講談社)などなど．

なかった女(ひと)が，こうものめり込んでしまうところに，人生の不思議，人との縁(というか数学との縁か)をみる思いがする．

システムは，MacintoshⅡfx，メモリ32Mバイト，ハードディスク1Gバイト2台と500Mバイト1台，光磁気ディスクはTahiti，純正16インチモニタ，OSは英語版System7，そしてTHINK C 5.0といった構成．

2,200×2,200のフルカラービットマップを扱うので，データ量が増えるとはいえ，ディスクが個人で2.5Gバイトだ．ものすごい時代になったものだ．

木本さんのパソコンとの付き合いは，

「Macintosh 128K*5を7～8年に前買ったのがMacとの付き合いのはじまりです」

最初は，MacPaintを使いはじめる．ここまでは普通のパターン．

しかし，

「MacPaintに限界を感じて，プログラミングに挑戦してみようと思って．最初はMS BASIC*6．これでプログラミングの練習をしました．雑誌に載っているプログラムリストを入力して，それをあれこれモディファイして，勉強していったんです」

MacPaintのあとにプログラミングへ方向を変える．ここが普通とはかなり違う．普通は，MacPaintに限界を感じたら，別のお絵かきソフトを探すものだ．プログラミングへとハンドルを切る人はめずらしい．

「そのころ，小説にも飽きていたし，ブルーバックスで，集合とか空間とかについて書かれた本をなんとなく買ったんです．それがきっかけみたいなもんですね．数学に興味をもつようになったのは」

講談社ブルーバックス編集部，泣いて喜びなさい．ブルーバックスは見事，その目標を達成しておりますぞ．

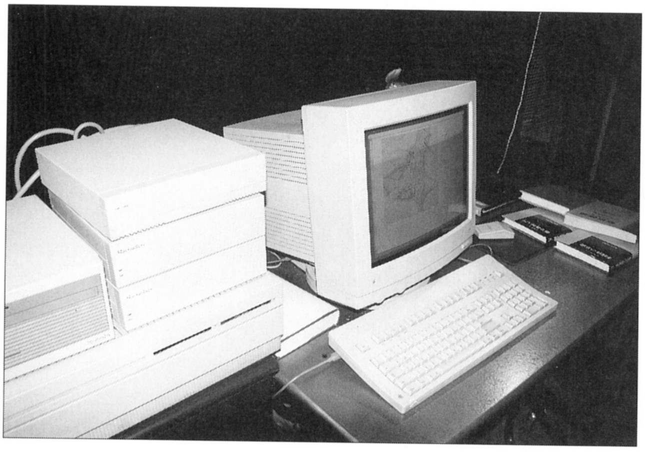
「数学は高校のころはまったくやってなかったし，その後もまるっきりだったんです．で，とりあえず，微積分から手をつけてみました．変化や無限という概念に，興奮しました」

手始めが，あなた，微積でっせ，微積．信じられますか．さらには，無限などという抽象概念に感情移入ができる．やはり木本さんは，数学的なものと波長が合っていたのだろう．暇をみつけては，コツコツと数学書と格闘する日々が続いた．

木本「それでも，1年半ほどは，なかなか数式の意味が分からず苦労しました」

中村「ぼくなんか，いまでも分からないけど(笑)．木本さんは，そのあとはすいすいと?」

木本　　「そんなことはありません．けれ



←木本さんのシステム．構成内容は本文にあるほか，ディスプレイはMacintosh16インチカラーディスプレイ，プリンタはLaserWriterⅡNTXを使っている．

*5 10年前に登場した，初代Macのこと．メモリが128K，CPUは68000の7.83MHz．フロッピーが1台で，ハードディスクはなし．これでも標準添付されていたMacPaint，MacWriteは，軽やかに動いた．はじめてみたときは，ほんとに感動的だった．とくに，MacPaintで描いたグラフィックがMacWriteの文書の中にクリップボード経由で貼り込めたときの驚きといったらなかった．

*6 もう知らない人のほうが多いだろうが，MicrosoftはMac用のBASICも出していた．

*7 そこまでの境地に達したことがないので，ほんとのところは分からない．それでも想像できることはあって，数学的記述方法は，無駄を省いてあるので，分かり始めると濃縮されたエッセンスを丸呑みするようなもので，非常にのめり込むだろうということ．ちょうど詩にのめり込むと，小説なんて無駄が多すぎて読む気にならないのといわれるが，それと似ているのではないか．

ど，形式化，抽象化されているおもしろさをだんだん感じるようになっていきました．特に，抽象化のすごさですね．抽象化のすごさには目のくらむ思いがしました」

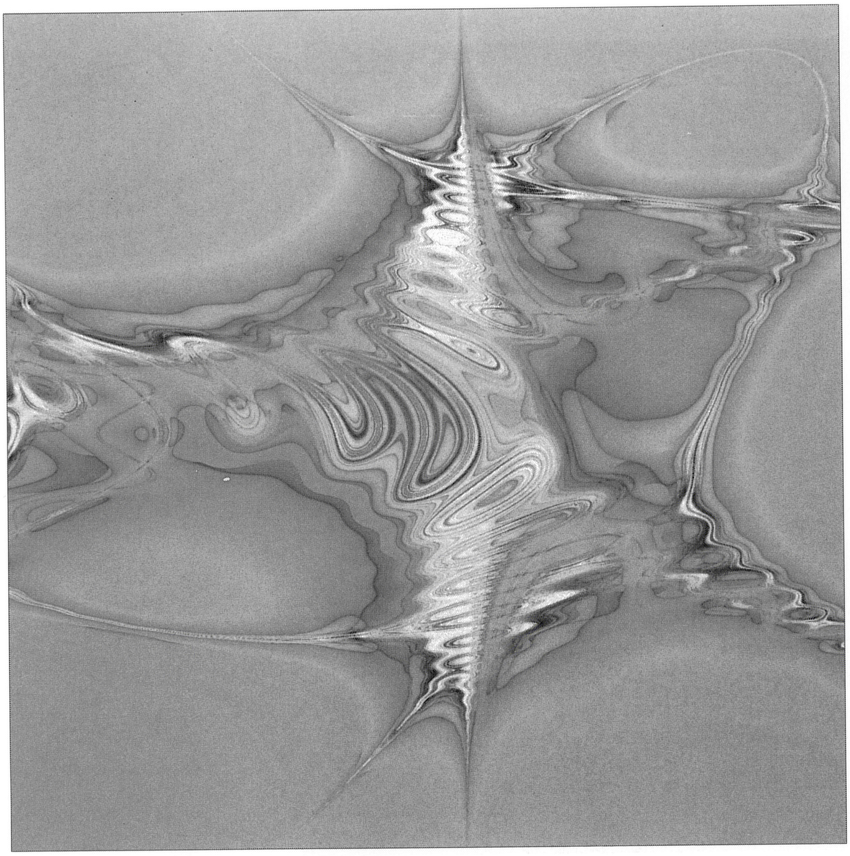
PostScript(!!)からTHINK Cへ

何事もそうだが，最初のハードルは高い．そのハードルをクリアするまで根気が続けば，あとはだんだんとおもしろさが分かってくる．こういう境地に達すると，数学書を読んでも，小説を読んだり映画を観ているよりもおもしろくなるはず*7．人間，努力だぜ．

中村「それでプログラミングを本格的にやるようになるんですか」

木本「そのころになると，評判のいいグラフィックツールも出てきたんですが，やっぱり半年ほどで飽きるんです」

中村「Mathematicaなどは使ったりはしなかったんですか」



「空間の状態，表情といったものに焦点があるため，いわゆる3Dソフトの手法が使えないので，自分なりに工夫して作ったプログラムによる作品の一つ．複素関数で極を多くもつものを作り，曲線座標上にその値を色価に変えて与えたもの．座標の曲がりに沿ってパターンが複雑に歪んで写る．仕組みは分かって描いているのに，実際にモニタ上にグラフィックが現れるまで，どのようなフォルムになるのかは，作者自身にも分からない．予想できるのは，せいぜい3割くらい」（木本氏）



木本圭子(きもと・けいこ) 1983年，多摩美術大学テキスタイルデザイン科卒業．雑誌・書籍などのデザインを手がける一方，数年前より，コンピュータを使った，数理的手法による“変容の形象”の実現を実験しはじめる．経験としての視覚対象から離れ，フォルムというものを一度抽象空間に押し上げて見直すことで，新たな視覚地平を探検中．

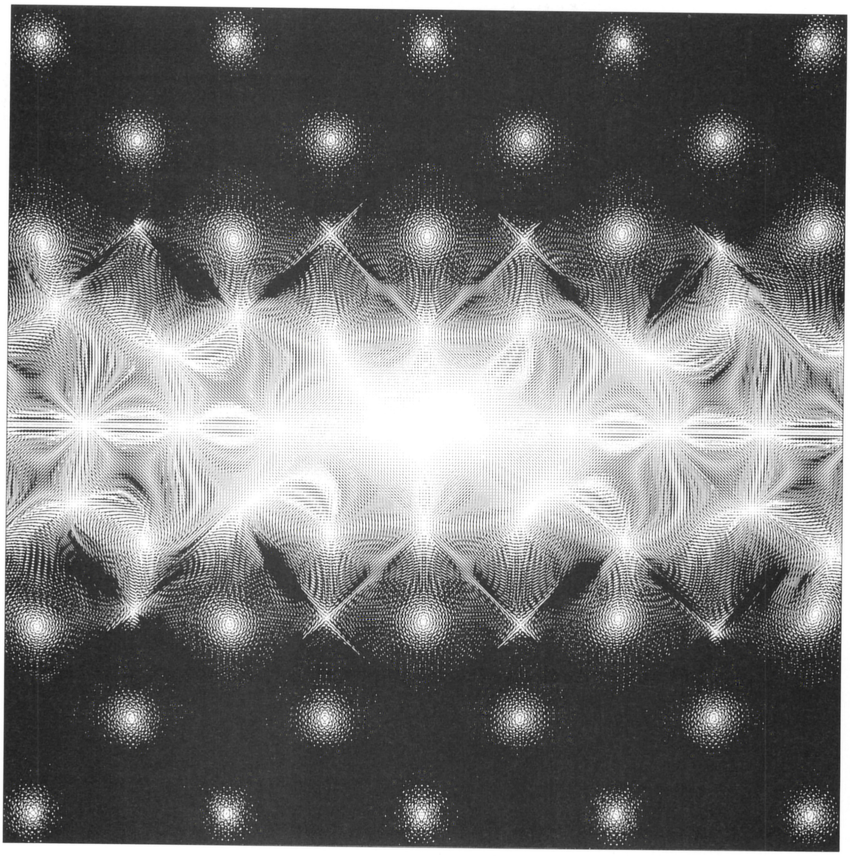
木本「Mathematicaなどは，固有値の計算や式の展開をやったり．プロトタイピングツールとしては使っていますね」

中村「グラフィックツールに飽きるのは，何か原因があるんでしょうかね」

木本「結局，自分の想像力の内側にとどまってしまう感じがするんです．方法の境界もはっきりしているし．プログラムで計算させると，何が飛び出すか分からない，そのおもしろさがありますね．プログラムによって関数を計算した結果を，視覚化するおもしろさですね．マンデルブロ集合のグラフィックをみたときの衝撃は，筆舌につくしがたいというか．あれは人間の手では決して描けない絵でしょう．想像力の地平を拡げてくれるパワーを感じたんです」

中村「ああ，なるほど．よくわかる，そういうことか．それでとうとうTHINK Cに挑戦と」

木本「いえ，Cでやる前は，PostScriptでプログラミングしてたんです」*8



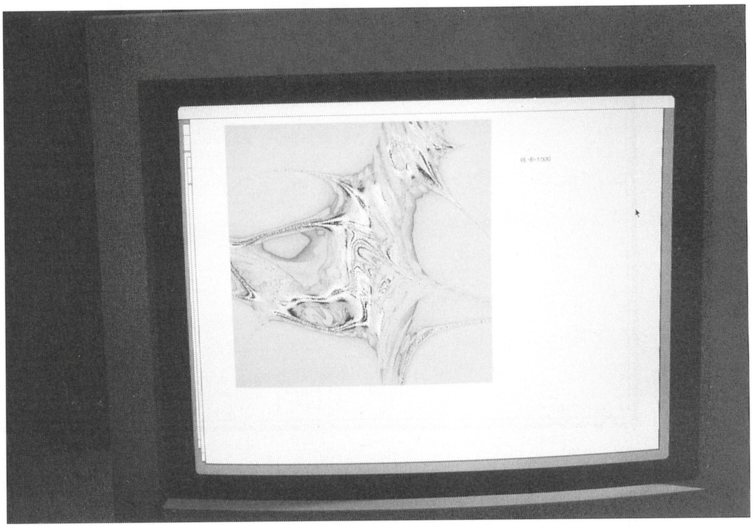
「PostScriptで制作していたころの作品．“物の何もない空間に，しかし性質がある”という“場”の考えに興奮して作ったものだが，まだ，場というには素朴すぎ，むしろベクトル関数というものにリアリティを感じはじめ，個々の関数のもつ性格に魅了されて描いていた．数学と記号の羅列にしか見えなかった数式が表現の言語なんだとはじめて身にしみたころ」(木本氏)

＊8 PostScriptは，ページ記述言語の事実上の標準．文字や図形をページ(紙や画面)上，どう割り付けるかをプログラム的に記述する．

＊9 すごい隠し玉があったものだ．中村は激しく動揺しつつ，喜んでおり，興奮を隠せない．

＊10 これだから，シロートは怖いと，あきれている．逆をいうなら，自分が表現したいものを持っている人は，ものおじしない．自分が表現したいものを持っている．これは強い．

＊11 PowerPCはIBM, Apple, Motorolaが共同開発した新型のCPU．Intelの次世代CPUであるPentiumより，スピード，熱発生，大きさ，値段などさまざまな面で優れており，動向が注目されている．木本さんの場合は，THINK CのPowerPC対応コンパイラさえ出れば，確実にプログラムは速くなるはず．浮動小数点演算が多いなら，10倍は速くなるのでは．



中村「えぇっ！ PostScriptですか．あなた，PostScriptなんて人間がプログラムする言語じゃないですよ＊9．プリンタドライバが知ってりゃいいんです」

木本「えっ，そうなのですか．PostScriptだとグラフィックが出せると聞いたものですから．PostScriptをやってみたかったし，それでプログラムを書いてPostScriptで出せる出力ショップにもっていって，動かしてもらいました．そうしたら，実行して出力されるまで1時間かかりました」

中村「すんごいなあ＊10．きっと，ループ，何百万回も回したんだろうなあ．普通，出力ショップで，やりませんよ，そんな怖いこと」

木本「まあ，そうでしょうね(笑)．そんなことがあって，言語的に弱いと思って知人に相談したら，Cにしたらって．MacのCならTHINK Cだと教わって移行することにしたんです」

中村「で，Macのプログラムをバリバリと」

木本「いえいえ，とんでもない．イベント処理とかって，メインループ以外の部分は，友人のプログラマに頼んで作ってもらいました．私は，ループ内の関数の計算のところをいじる感じですね」

マシンは最初のMacintosh 128Kから，Macintosh 512K，II，そして現在のIIfxと変わった．

「今度MacintoshのAVシリーズが出たので，今後は動画もやってみたいなと思ってます．いま，なによりほしいのがスピード．来年，PowerPCになって速くなるといいんですが」＊11

3年前に現在の手法でやってみようと決めた．しかし，1年半ほど暗中模索の日々が続いたという．

「独学だから効率が悪いし，それに系統立って勉強してないから，筋が悪いんじゃないか．そんな不安も感じています．それでも，経験を積んだから，最近はかなりあたりがついて，以前ほどには迷わないようになりました」

●

いま，彼女は，暇をみつけてはコツコツと作品作りに励んでいる．

数学と無縁だった女(ひと)が，一歩一歩，ここまで積み上げてきた．ぼくが数学に劣等感を抱いていることを割り引いても，ぼくは，これ，すばらしいと思った．

ぼくは彼女に勇気づけられて，少しはむずかしい本でも読もう，数学の本でも読もうと思った．そして，案の定，まだ思っただけである．

でも，何か，あなたも今日から始められるはずだ．**何事も，死ぬまでは，遅すぎるということはないのだから．**

(次回は東京大学理学部助教授・坂村健氏の予定です)



「30行計画」にも対応!!

コプロセッサの
レジスタ内容を表示する

SYMDEB.EXE

適応機種●MS-DOSマシン
言語●Quick BASIC Ver.4.5
適応ソフト●SYMDEB.EXE（Ver.3.01）
著者●安田充久 *Mitsuhisa YASUDA*

はじめに

パソコンユーザにとって，SYMDEBはとてもありがたい存在です．もちろん，高級言語で書いたプログラムをデバッグするためならば，今では，第3世代のデバッガCodeViewやTurbo Debuggerが使えます．ソースレベルのデバッグができ，386命令にも対応しているのはとても便利です．けれども，作業の種類によっては，第2世代のデバッガSYMDEBの方が，小回りがきいて便利なこともあります．

筆者は，この便利なSYMDEBをさらに使いやすくしようという仕事に，ここ数年断続的に取り組んできました．**参考文献2，3**はその成果です．このテーマを取り上げるときに筆者にとって忘れられないのが，堀英明氏による漢字対応(**参考文献1**)です．「デバッガにパッチを当てる」という発想が，当時の筆者には驚きでした．冷静になって考えれば，デバッガといえどもひとつのソフトウェアに過ぎませんが，デバッガの動作は何となく神秘的で，これだけは別格の聖域という思い込みがあったのです．

SYMDEBの改造といっても，別段計画的にやっているわけではありません．むしろ場当たりといった方が正確でしょう．というわけで，今回は，SYMDEBのコプロセッサ対応を取り上げることにしました．また，これまでの改造内容も合わせて，全体をひとつのファイルにまとめました．同時に，既存の拡張機能については全面的に見直しを行い，細かい改良を加えました．特に，画面切り替え機能は，「30行計画」にも対応しました．ついでといっては堀氏に失礼ですが，漢字対応も若干の改良を加えて取り込ませていただきました．

改造の方法

改造の対象とするのは，NEC版のMS-DOSに付属するSYMDEB.EXE(Ver.3.01)です．NEC版MS-DOSに付

属するSYMDEB.EXEは，これまでのところ，どれもVer.3.01であり，タイムスタンプが異なっていますが，ファイルの中身は同じです(MS-DOS 5.0Aの拡張機能セットに含まれるSYMDEB.EXEは少しだけ中身が違いますが，これにも対応しています)．絶対番地を参照してパッチを加えるので，これ以外のバージョンはサポートしていません．バージョン番号は，SYMDEBを起動したときに画面に表示されます．

前回までとは異なり，オリジナルのSYMDEB.EXEを必要とします．**参考文献1〜3**により改造されたファイルを入力ファイルとすることもできますが，これまでの改造はすべてクリアされるので，オリジナルのSYMDEB.EXEを使うのと同じになります．特に重要な注意として，本ユーティリティにより改造されたファイルを入力ファイルとすることはできません．したがって，改造済みファイルのINT 21Hコメントを変更したい場合にも，オリジナルのSYMDEB.EXEを入力ファイルとして使用して下さい．

SYMDEBを改造するためのプログラムSDEB.BASは，QuickBASICにより書かれています．改造の方法は簡単です．以下の手順に従って下さい．

①QuickBASICを

QB /L ↵

により起動します．スイッチ/Lをつけるのは，デフォルトのQuickライブラリQB.QLBを使用する必要があるからです．
②カレントドライブにオリジナルのSYMDEB.EXEを置きます．
③プログラムSDEB.BASを実行します．

以上の手順により，改造されたファイルSDEB.EXEがカレントドライブに作成されます．SYMDEB.EXEはもとのままです．ディスプレイに

改造作業を終了しました．

と表示が出れば，正常終了です．このメッセージが出ない場合には，改造作業が失敗です．SYMDEB.EXEがオリジナルのVer.3.01であれば，失敗することはありません．

でき上がったファイルは，元のファイルより長くなります．長さは，INT 21Hのコメントの分量に依存します．デフォルトで実行した場合には，ファイルの長さは39,212バイトになります．オリジナルのSYMDEB.EXEは36,538バイトですから，差し引き2,674バイトだけ太りました．

拡張された機能

この改造によりサポートされる機能の一覧を以下に示します．このうち，❶〜❸が今回新たに取り上げるものです．漢字対応機能❹は，**参考文献1**によるものであり，❺は**参考文献2**で，❻〜⓫は**参考文献3**で既に取り上げました．本稿では，❹〜❽の改良に触れています．❾〜⓫の機能には変更がありません．

❶7コマンドが新設され，8087レジスタの内容を表示することができます．
❷Aコマンドでの FSTP命令のアセンブルのバグを修正します．
❸MS-DOS Ver.5に対応します．
❹DA，DBコマンドで漢字によるダンプ表示が行われます．漢字対応機能は，キー操作により停止することもできるようにしました．
❺Jコマンドが新設され，これにより分岐命令の検索を行うことができます．Yコマンドが新設され，これによりCALL命令の検索を行うことができます．
❻ブレークポイントの動作を変更します．
❼INT 21hファンクションに対するコメントを拡張します．今回は，新たにサブファンクションもサポートしました．
❽画面の切り替えをNEC機でも可能にします．今回は，「30行計画」にも対応しました．
❾80287命令FSTSW AXをサポートします．
❿DL，DTコマンドのバグを修正します．
⓫EXEPACKされたファイルもそのままデバッグ可能にします．

ここで，これらの機能の機種依存に触れておきましょう．SYMDEBは，もともとMS-DOS汎用に作られているので，MS-DOS上ならば，どんな機種でも動くはずです．上記の拡張機能も，❽の画面切り替え機能を除けば，機種依存はありません．

画面切り替え機能は，ハードウェアに依存して実現されており，SYMDEBは，本来この機能をIBMマシンだ

けで実現しています．本プログラムにより改造されたSYMDEBでは，IBMマシンとNECマシンで画面切り替え機能が使えます．その他の機種では，この機能は使用できません．

❶8087レジスタ内容の表示

SYMDEBは，コプロセッサ8087の命令をサポートしていますが，プログラムをデバッグする過程で8087レジスタの内容を表示することはできません．したがって，今レジスタにどんな数値が入っているかを知ることはできません．メモリにストアして，ダンプコマンドを使用して初めてどんな数値が入っていたかを知ることができます．

これではデバッグの作業に不都合なので，8086レジスタと同様に8087レジスタの現在の内容も表示できるようにしました．この機能は，SYMDEBの起動時にスイッチ/7を指定すると有効になります．

コマンドとしては，CodeViewと同様に"7"を割り当てています．表示される内容は，CodeViewのように馬鹿丁寧ではありませんが，デバッグの目的にはこれで十分でしょう．たとえばST(0)レジスタにπの値が入っているとすると，7コマンドを

```
-7↵
```

と入力することにより

```
ST(0)=+3.14159265358979324 E+0000
```

という表示が出ます．数値の入っていないレジスタは

```
ST(1)=empty
```

と表示されます．

レジスタの内容によっては，表示が

```
NaN(Not a Number)
unnormal
denormal
infinity
```

となります．CodeViewの7コマンドとは違って，空のレジスタも含めて8個のレジスタの内容がすべて表示されます．ステータスワード，コントロールワードの表示はありません．

このダンプには，8087のFBSTP命令を活用しました．その結果，上の例で分かるように，10進18桁の詳しい表示が可能になりました．ちなみに，DTコマンドでは16桁しか表示されません．2進10進変換は，基本的にはFBSTP命令が引き受け，レジスタ内の整数をパック十進数として出力してくれますが，それでもいくらか注意してプログラムを作りました．浮動小数点数の出力というのは案外に厄介なもので，レジスタ内部では正確に2.0という数値が画面では，

```
1.99999……99
```

と表示されることが一般によくあります．ここでは，そういうことがなるべく起きないように注意したつもりです．

7コマンドは，もちろん，コプロセッサつきのマシンだけで有効です．起動スイッチに/7が指定してあると，起動直後にコプロセッサの有無を調べ，コプロセッサがない場合には7コマンドが動作しないようにしてあります．

ここまでできると，Rコマンドと同様に，レジスタの値を変更したいという希望が出てくるかもしれません．FBLD命令のお世話になれば不可能ではありませんが，そこまでやるのは面倒なのであきらめました．普通のデバッグでは，コプロセッサを使うとしてもこの程度で十分でしょう．

なお，7コマンドを使うと，画面の表示が繁雑になります．したがって，SYMDEBの起動時にスイッチ/Sも指定して2本立ての画面にするのがよいでしょう．

7コマンドの使い方として，これを単独に使うのは得策とは言えません．マルチコマンド機能を利用して，たとえば

```
-T;U;7↵
```

のように一括して入力すると，デバッグ作業がスムーズに進みます．VZエディタが常駐していると，このようなキー入力を繰り返すのは容易です．

FSTP命令のアセンブルの問題

以上のようにして7コマンドが組み込まれたので，そのテストを兼ねて，いろいろのコプロセッサ命令を実行してみました．こうして7コマンドが使えると，コプロ

セッサの動作がとてもよく理解できます．コプロセッサに特有のスタック動作は，図に描いて説明すると何枚もの図を必要としますが，このように実際の動作を追いかけていけば，初めての人もすぐに頭に入るでしょう．8個のレジスタがいつも目に見えるのが分かりやすさの原因です．コプロセッサの仕組みをこれから勉強したいという人にも便利な機能だと思います．

ところで，このテストの最中に，SYMDEBのAコマンドがFSTP命令をアセンブルするときにバグを発生することが判明しました．バグの内容は，ニーモニック

```
FSTP    ST(1)
```

のコードを

```
D9D9
```

にアセンブルするというものです．387ではこれでも問題ないようですが，筆者の愛機の8087は，このコードに対して

```
FSTP    ST(0)
```

と同様に動作します．すなわち，ST(0)が単にPOPされるだけになっています．上記命令の使用は，コプロセッサを使うプログラムでは珍しくありません．たとえば，本プログラムでも，10のベキ乗を計算するところで使っています．そこでのコードは，プログラムにあるように

```
DDD9
```

です．筆者の8087も，これならば正しい動作をします．このコードは，MASMがアセンブルした結果です．Intelのマニュアルを調べると，表向きのコードは“DDD9”だが，“D9D9”でもよいように注がつけてあって，8087のチップのバグなのか，それともSYMDEBのバグなのかハッキリしません．しかし，どちらにしても動作が不正なことに変わりはないので，MASMのしていることが正しいのなら，これはSYMDEBのアセンブルのバグであると考え，SYMDEBに1バイトの修正を加えました．

なお，動作に不具合があるのは，レジスタにFSTPする場合だけであり，メモリにFSTPする命令は，DWORD，QWORD，TBYTEのいずれも問題ありません．上記の1バイト修正は，前者のコードを変更しますが，後者のコードには影響しません．

❸MS-DOS Ver.5への対応

DOS5(MS-DOS Ver.5)では，バージョン番号取得ファンクション(AH=30h)が少し変更されました．このファンクションはBHレジスタにOEM番号を返す仕様になっています．ところが，DOS5でOEM番号を知るためには，AX=3000hとする必要があります(**参考文献4**)．

項目❽に関連して**参考文献3**で説明したように，SYMDEBにとって，OEM番号は極めて重要な意味を持っています．すなわち，2本立て画面の実現はMS-DOSのファンクションだけでは不可能であり，機種を判別してハードウェア依存コードを使用する必要があります．SYMDEBは，全体としてMS-DOS汎用に作られていますが，この部分だけは機種依存するように作られており，機種判別のためにOEM番号を利用しています．したがって，OEM番号は正確であることが必要です．そこで，SYMDEBをDOS5に対応するよう変更を加えました．

SYMDEBは，これ以外にもMS-DOSのファンクションを多用しています．当然のことながら，非公開のファンクションも使用しています．しかし，DPB(Drive Parameter Block)を参照するというようなことはしていないので，DOS5対応で必要な変更は，これひとつだけで済むようです．

正確に言うと，SYMDEBの中では，DPBポインタ取得ファンクション(AH=32h)が1回だけ使用されています．ただ，そこではDPBアドレスを取得することが目的ではなく，単にドライブが有効かどうかを判定するのに使われています．したがって，DOS5のファンクションと抵触しません．

なお，NEC版のMS-DOS Ver.5.0Aに付属するSYMDEB.EXE Ver.3.01には，バージョン取得ファンクションの仕様変更に対応した修正が施されています(それ以外の変更は加えられていません)．したがって，Ver.5.0Aに付属のSYMDEB.EXEに対しては，本項目のパッチを加えないようにしてあります．

❹DA，DBコマンドでの漢字対応

日本語対応MS-DOSでは漢字を扱うことができますが，SYMDEBのダンプコマンドは，漢字表示に対応し

ていません．漢字表示を可能にするのが，堀英明氏によるパッチあてです(**参考文献1**)．筆者は長い間，この恩恵にあずかってきました．**参考文献1**は今ではかなり古い記事であり，今回が最終版のつもりなので，この便利な機能を取り込ませていただきました．堀氏のプログラムは外字に対応していないので，外字も表示できるよう変更してあります．

ところで，漢字対応は確かに便利ですが，いつでも漢字対応がよいかというとそうとは限りません．漢字対応になっていない方がダンプリストが見やすい場合もあります．たとえば，00h～7Fhのアスキー文字列を沢山含んだファイルを用意するとき，区切り記号として00Hを使うのがひとつの方法です．しかし，もう一つの方法として，最後の文字に80hを加えて区切り記号とする方法があります．データの個数が非常に多い場合には，これによってバイト数を節約することができます．

実例を挙げると，SYMDEBがアセンブルに使用しているニモニックの一覧表(CS:74B8番地以降)は，このような構造になっています．こういう場合には，非漢字対応の方が便利です．

この対策として，漢字対応版と非漢字対応版の2種類のSYMDEBを用意することが考えられますが，いちいち使い替えるのは面倒です．そこで，[CTRL]+[K]を入力することにより，漢字モード/非漢字モードの切り替えができるようにしました．デフォルトのモードは漢字モードにしてあります(非漢字モードをデフォルトのモードにすることも可能です)．VZエディタを常駐している場合には，[CTRL]+[P]+[K]と入力します．

[CTRL]+[K]の後に[↵]キーを押すことを忘れないで下さい．もっとも，SYMDEBはセミコロンで区切ったマルチコマンドを許していますから，

－^K; DB CS:74B8 [↵]

のようにまとめて入力することもできます．

❺J，Yコマンドによる分岐命令/CALL命令の検索

あるアドレスのところへ，分岐命令によりどこから跳んでくるかを調べるのがJコマンドです．また，あるサブルーチンが，どこから呼び出されるかを調べるのがYコマンドです．これらのコマンドは，ソフトウェアの動作を分析するときに威力を発揮します．

どちらのコマンドも**参考文献2**で取り上げたものですが，ここでは，Jコマンドに小さな改良を加えました．すなわち，検索の対象とする分岐命令として

LOOP，LOOPZ，LOOPNZ

の3個を加えました．結局，Jコマンドにより検索される分岐命令は

JMP(Near Jump命令，E9XXXX)
JMP SHORT(EBXX)
JCXZ
JO，JNO，JZ，JNZ，JA，JB，JAE，JBE
JS，JNS，JP，JNP，JG，JL，JGE，JLE

を含めて，合計22個です．386以上のプロセッサに固有の分岐命令には対応していません．

一方，Yコマンドはセグメント内コール命令(E8xxxx)を検索します．どちらの場合も，アドレスの即値がプログラム中に書かれているわけではないので，Sコマンドにより検索することはできません．

どちらのコマンドも，デフォルトでは，CSセグメントが検索の対象となります．

❻ブレークポイントの動作を変更する

SYMDEBによりプログラムのデバッグを行うと，暴走することがあります．この暴走を少なくして，なるべく安定してSYMDEBを動作させたいというのが，この項目の目的です．技術的な詳しい説明は，**参考文献3**を参照して下さい．

ここでは，**参考文献3**からの変更について説明しましょう．BPの状態をdisabledに変更する条件を厳しくしました．すなわち，ブレークポイントでの1バイトが0CCHから変更された場合でも，それがもともとのコードと同じ場合には，BPの状態をabledのままとし，disabledに変更するのを止めました．この結果，オーバーレイなどによりコードの書き換えが行われても，前と同一のコードの場合にはBPの状態がabledのままに保たれるようになりました．

この小さな変更は，『MS-DOSウルトラユーティリティーズ』(技術評論社)に収録されたSDEB.BASでは既に済んでいます．

⑦INT 21hファンクションに対するコメント表示

SYMDEB上で1ステップずつ命令を追いかけていくと，INT 21hを実行する直前のところで，AHレジスタの値に従って，ファンクションの内容が短いコメントとして表示されます．

参考文献3では，SYMDEBがコメントを用意していないファンクション(その多くは非公開ファンクション)について，コメントを表示できるように機能拡張をしました．今回は，さらにこの機能を拡充して，次の二つのことを可能にしました．

①SYMDEBがコメントを内蔵しているファンクションについても，新たに別のコメントをカスタマイズして表示させることができる．
②サブファンクションごとにキメの細かいコメント表示を可能にする．

この動機としては，DOSのバージョンが上がるにつれて，AHレジスタによるファンクション指定だけでは不十分になり，ALレジスタによるサブファンクションの指定が増えてきたという事情が挙げられます．AHだけでコメントを共通にすると，ALがさまざまの値を取った場合に，その意味が理解しにくいファンクションが出てきました．極端な例はIOCTLファンクション(AH=44h)です．これには，AL=00hから11hまで18個ものファンクションが属しています．

コメントは，AH=00h～7Fhの任意のファンクションについて表示させることができます．コメントの設定の仕方は，実に簡単です．プログラムの末尾がコメントのリストになっているので，ここにファンクション番号とコメントをDATA文として書き並べればよいのです．ここに指定しないファンクションについては，もしもSYMDEBの中にあらかじめコメントが用意されていれば，それが表示されます．用意されていなければ，コメントは何も出ません．

コメントの書き並べ方は，プログラムを見ればすぐに分かるはずですが，念のために規則のようなものを書いておきましょう．

①順序は，AHについては任意です．ただし，昇順にしておく方が分かりやすいでしょう．
②サブファンクションALの指定がある場合は，AHの等しいものをまとめて書きます．ALについて昇順になっている必要はありませんが，昇順の方が見やすいでしょう．
③コメントは，アスキー文字を使って書きます．24文字までで打ち切られます．
④最後に80h以上の数字を書いておきます．これがコメントデータの終了マークです．

これだけでほとんど誤解の余地はないと思いますが，今回はサブファンクションまでサポートすることにしたので，少していねいに説明しておきます．たとえば，コメントのリストが

```
3700, "Get Switch Char DL"
3701, "Set Switch Char DL"
```

となっていると，AX=3700hと3701hではこの二つの異なったコメントが出ます．ありえないことですが，AX=3702hの場合には，何もコメントがでません．もしもこれを

```
37, "Get(0)/Set(1)SWITCHARDL"
```

としておくと，AH=37hならば，ALがいくつであってもこのコメントが表示されます．

また，

```
4B00, "Load & Execute Program"
4B03, "Load Overlay Program"
```

としておくと，4B00hと4B03hファンクションについてそれぞれこのようなコメントが出ます．この場合，AH=4Bhであって，ALが0，3以外のときは，コメントが何も出ないことに注意してください．

◆

以上の規則に従って，コメントを自由に設定して使用して下さい．コメントが増えると，ファイルは長くなりますが，それに応じてスタック領域を後退するようにしてありますから，心配はいりません．ファイルサイズを気にしなければ，かなりたくさんのコメントを設定できるはずです．

❽画面切り替え機能

画面切り替え機能は，起動時スイッチ/Sにより有効になります．これを使うとデバッグの作業が大変分かりやすくなります．たぶん，一度使ったらやめられないでしょう．

いささか極端な例ですが，本稿の場合，

SDEB /S SDEB.EXE ↵

によりプログラムの虫出しを行っています．このように2階建てでデバッガを走らせると，今，画面に出ているプロンプトがどちらのSDEBから出ているのか判断に迷うときがあります．迷ったときには，迷わず

ー¥↵

とします．親のSDEBにはスイッチ/Sを付けてあるので，もしも画面の切り替えが起きれば，親がプロンプトを出していることが分かります．もしも子のSDEBがプロンプトを出しているのだったら，エラーとなり，画面の切り替えは起こりません．このようにして，今1階にいるのか2階にいるのかを判別することができます．

この画面切り替え機能は，ノーマルモード/ハイレゾモードのどちらでも共通に使用できます．今回は，これを**参考文献5，6**の「30行計画」に対応させました．対応の方法としては，30行BIOSを呼び出すのではなく，MS-DOSのワークエリアから直接に行数を読み出しています．同時に行モードも読み出しているので，結果として，「行間空きモード」にも対応しています．

参考文献3にも書きましたが，/Sスイッチを指定した場合，SYMDEBは16Kバイト×2=32Kバイトのメモリを自分自身のコードの直後に確保する仕様になっています．したがって，原理的には，51行まで画面切り替えが可能です．実際に筆者が確認したのは30行までです．

その他の機能

項目❾～⓫には変更がありません．

おわりに

ここまで原稿を書いてきて，あらためて，本稿冒頭の項目リストを眺めてみました．わずか2,674バイトの増加でこれだけの機能拡充ができたことに，今さらながら驚きました．

こんなふうに機能の拡充ができることが分かると，もっといろいろの機能を盛り込みたいという気持ちが働きます．たとえば，INT 21hに詳しいコメントが自由につけられるのだったら，他の割り込みにもコメントをつけられるはずです．BIOSにコメントをつけるのは機種依存になりますが，EMMファンクションにコメントを付けるという程度なら，手ごろな作業です．

さらに手を広げるなら，本文に書いたようにコプロセッサのレジスタ内容を変更できるようにするというテーマもあります．また，386のレジスタを表示するとか，32ビット命令までアセンブル/逆アセンブルさせるという課題も考えられます．他にもまだあるでしょう．

けれども，全体的なバランスからいうと，今くらいがちょうど適当なように思います．SYMDEBのメリットはサイズが小さいという点にあり，小さいファイルから大きいファイルまで扱うことができます．サイズを大きくして機能を増やせば，第3世代のデバッガとあまり違いがなくなり，SYMDEBの意味がかえって薄れてしまうでしょう．

〈参考文献〉

1)堀英明，『漢字対応SYMDEB.EXE』，The BASIC'86年12月号
2)安田充久，『ジャンプ命令もサーチするSYMDEB.EXE』，The BASIC'89年7月号
3)安田充久，『MS-DOSの非公開ファンクションもコメント表示するSYMDEB.EXEの機能拡張』，The BASIC'92年4月号
4)イニシャルはN.N.，『MS-DOS 5.0のプログラミング』，ざべ'92年5月号
5)hironon，『PC-9801で30行を表示する試み』，ざべ'92年7月号
6)古庄歩，國井淳，『30行計画を統合しよう』，ざべ'93年7月号

List1 SDEB.BAS

```
'-------------------------------------------------------------
'  "SDEB.BAS" in QuickBASIC
'    MS-DOS 3.1/3.3のデバッガ SYMDEB.EXE (Ver 3.01) の機能を拡張する.
'-------------------------------------------------------------
' 1. ブレークポイントでのコード復元動作を改良する.
' 2. コプロセッサ命令 FSTSW AX をアセンブル／逆アセンブル可能にする.
' 3. DL/DT コマンドにより表示される実数の指数部の値が不正な場合が
'     あるので, 修正する.
' 4. 画面切り替え機能を PC-98 でもサポートする.
'     この機能は,「30行計画」にも対応している.
' 5. EXEPACK された EXE形式のファイルもデバッグ可能とする.
' 6. INT 21h ファンクションコールを実行したときに表示されるコメントを
'     非公開ファンクションにも拡張する.
' 7. MS-DOS Ver. 5.0 に対応.
' 8. DB,DAコマンドでの漢字対応 (原作は, 堀英明氏による).
' 9. J,Yコマンドを追加する.
'     J コマンド: 分岐命令によるアドレス参照を検索する.
'     Y コマンド: call命令によるアドレス参照を検索する.
'10. 7コマンドを追加する.
'     7 コマンド: 8087の8個のレジスタ値を表示する.
'11. FSTP命令のアセンブルのバグを修正する.
'     Ver 2(final), by M. Yasuda, 1991-1993.
'.............................................................
' << 参考文献 >>
'堀  英明:『漢字対応 SYMDEB.EXE』 "The BASIC", '86年12月号
'安田充久:『ジャンプ命令もサーチするSYMDEB.EXE』"The BASIC",'89年7月号
'安田充久:『MS-DOSの非公開ファンクションもコメント表示する
'            SYMDEB.EXEの機能拡張』"The BASIC",'92年4月号
'.............................................................
' 注意:本プログラム中のアドレスは, SYMDEB.EXE をそのままデバッガ
'        でオープンしたときのCSセグメントでのオフセットアドレスです。
'-------------------------------------------------------------
 DEFINT A-Z                         ' 変数は, 原則として全て整数型である.
 TYPE Regs
     AX    AS INTEGER
     BX    AS INTEGER
     CX    AS INTEGER
     DX    AS INTEGER
     BP    AS INTEGER
     SI    AS INTEGER
     DI    AS INTEGER
     flags AS INTEGER
     DS    AS INTEGER
     ES    AS INTEGER
 END TYPE

 DECLARE SUB InterruptX (num AS INTEGER, reg1 AS Regs, reg2 AS Regs)

 DIM inreg AS Regs, outreg AS Regs
 DIM buffer.area(&H5000) AS INTEGER      ' ゼロに初期化される.

 DEF fnPeekW (X)
   Z& = PEEK(X) + PEEK(X + 1) * 256&
   IF Z& > &H7FFF THEN
     fnPeekW = Z& - &H10000
   ELSE
     fnPeekW = Z&
   END IF
 END DEF

 CLS
 PRINT SPC(15); "SYMDEB.EXE 改造ユーティリティ"
 PRINT SPC(50); " -- M. Yasuda (1991-1993)"
 Fin$ = "SYMDEB.EXE"                  ' 入力ファイル名
 Fout$ = "SDEB.EXE"                   ' 出力ファイル名

 segm = VARSEG(buffer.area(0))        ' 作業エリア
 ofst = VARPTR(buffer.area(0))
 buffer.segm = segm + ofst ¥ 16 + 1
 buffer.ofst = 0

 GOSUB Load.bin                       ' ファイル読み込み
 PRINT "  入力ファイルの長さ ="; L&; "バイト"
 Load.segm = buffer.segm
 Code.segm = Load.segm + &H20         ' コード セグメント

 DEF SEG = Code.segm
 GOSUB Check.Stack                    ' バージョン検査.
 GOSUB Change.BP                      ' ブレークポイント処理
 GOSUB Enable.FSTSW.AX                ' FSTSW AX を可能にする.
 GOSUB Fix.DL.DT                      ' DL,DT のバグ修正.
 GOSUB Enable.Screen.Flip             ' /S スイッチを 98 でもサポートする.
 GOSUB Packed.Files                   ' EXEPACKされたファイルもデバッグ可.
 GOSUB DOS.5                          ' MS-DOS 5.0 に対応.
 GOSUB KNJI                           ' 漢字対応化 (堀英明氏による).
 GOSUB JANDEB                         ' J, Y コマンドをサポートする.
 GOSUB Dump.Reg87                     ' 7コマンドをサポートする.
 GOSUB Fix.FSTP                       ' FSTP命令のアセンブルのバグ修正.
 GOSUB Int21.Comments                 ' INT 21h コメント表示を拡張.
 GOSUB Change.Stack.Area              ' スタック領域を後退させる.
 GOSUB Write.Header                   ' ヘッダ書き換え
                                      ' ファイル書き出し
 L& = FileLength&
 GOSUB Save.bin
 PRINT "  出力ファイルの長さ ="; L&; "バイト"
 PRINT "改造作業を終了しました。"
 END
'-------------------------------------------------------------
ReadFileError:
 BEEP
 PRINT "Read File Error!"
 END
WriteFileError:
 BEEP
 PRINT "Write File Error!"
 END
'-------------------------------------------------------------
' ファイルを読み込むサブルーチン
' ファイル名: Fin$
' バッファアドレス: buffer.segm: buffer.ofst
' 読み込まれたファイルの長さ: L&
'.............................................................
Load.bin:
 F$ = Fin$ + CHR$(0)
 inreg.AX = &H3D00                    ' open handle
 inreg.DS = VARSEG(F$)                ' file name
 DEF SEG = inreg.DS
 inreg.DX = fnPeekW(VARPTR(F$) + 2)
 CALL InterruptX(&H21, inreg, outreg)
 IF outreg.flags AND 1 THEN GOTO ReadFileError

 inreg.BX = outreg.AX                 ' handle
 inreg.AX = &H3F00                    ' read handle
 inreg.DS = buffer.segm               ' buffer address
 inreg.DX = buffer.ofst
 inreg.CX = &HFFFF                    ' data length to be read
 CALL InterruptX(&H21, inreg, outreg)
 L& = outreg.AX                       ' file length
 IF L& < 0 THEN L& = L& + &H10000
 IF outreg.flags AND 1 THEN GOTO ReadFileError

 inreg.AX = &H3E00                    ' close handle
 CALL InterruptX(&H21, inreg, outreg)
```

```
RETURN
'----------------------------------------------------------------
' ファイルを書き出すサブルーチン
' ファイル名: Fout$
' ファイルの長さ: L&
' バッファアドレス: buffer.segm: buffer.ofst
'................................................................
Save.bin:
  F$ = Fout$ + CHR$(0)
  inreg.AX = &H3C00                      ' create handle
  inreg.DS = VARSEG(F$)                  ' file name
  DEF SEG = inreg.DS
  inreg.DX = fnPeekW(VARPTR(F$) + 2)
  inreg.CX = &H20                        ' file attribute
  CALL InterruptX(&H21, inreg, outreg)
  IF outreg.flags AND 1 THEN GOTO WriteFileError

  inreg.AX = &H4000                      ' write handle
  inreg.BX = outreg.AX                   ' handle
  inreg.DS = buffer.segm                 ' buffer address
  inreg.DX = buffer.ofst
  IF L& > &H7FFF THEN
    inreg.CX = L& - &H10000
  ELSE
    inreg.CX = L&                        ' file length
  END IF
  CALL InterruptX(&H21, inreg, outreg)
  IF outreg.flags AND 1 THEN GOTO WriteFileError

  inreg.AX = &H3E00                      ' close handle
  CALL InterruptX(&H21, inreg, outreg)
RETURN
'----------------------------------------------------------------
' ヘッダを書き換える.
'................................................................
Write.Header:
  POKE DataArea, &H1A                    ' ファイル終了マーク
  DataArea = DataArea + 1
  DEF SEG = Load.segm
  IF DataArea < 0 THEN
    FileLength& = DataArea + &H10200
  ELSE
    FileLength& = DataArea + &H200       ' ヘッダの長さ=&H200バイト
  END IF
  Page = FileLength& ¥ &H200             ' ページ数
  Res = FileLength& MOD &H200            ' 最終ページのバイト数
  IF Res THEN Page = Page + 1
  Adr = 2: V = Res:  GOSUB PokeW
  Adr = 4: V = Page: GOSUB PokeW
  Adr = &HE:  V = 0: GOSUB PokeW         ' スタックセグメント
  Adr = &H10: V = SP.value: GOSUB PokeW  ' スタックポインタ初期値
  PRINT "  ヘッダを書き換えました."
  RETURN
'----------------------------------------------------------------
Err.Data:
  PRINT "ＤＡＴＡ文に入力ミスがあります。"
  PRINT "ADDRESS="; HEX$(Adr)
  END
Err.Version:
  PRINT "SYMDEB.EXE のバージョンが違います。"
  END

'           ADR番地に2バイト整数 V を書き込むサブルーチン.
PokeW:
  Low = ASC(LEFT$(MKI$(V), 1))
  High = ASC(RIGHT$(MKI$(V), 1))
  POKE Adr, Low                          ' 第1バイト
  POKE Adr + 1, High                     ' 第2バイト
  RETURN

'           コードを書き込むサブルーチン.
Write.Code:
  READ A$
  Adr = VAL("&H" + A$)                   ' 書き込み開始アドレス
  DO
    READ D$                              ' 1行のコードを読む.
    IF D$ = "$" THEN EXIT DO             ' "$" は終了マーク.
    L = LEN(D$): IF L AND 1 THEN GOTO Err.Data
    FOR I = 1 TO L STEP 2
      POKE Adr, VAL("&H" + MID$(D$, I, 2))  ' 1バイトづつ書き込む.
      Adr = Adr + 1
    NEXT
  LOOP
  RETURN

'           ADR番地以降に文字列 M$ を書き込むサブルーチン
Poke.Str:
  FOR I = 1 TO LEN(M$)
    POKE Adr, ASC(MID$(M$, I))
    Adr = Adr + 1
  NEXT
  RETURN
'----------------------------------------------------------------
' SYMDEB プログラム中では，ＳＰレジスタを初期化するときに
' 即値が使われている。これらの8個の即値がすべて一致すれば
' バージョンに違いはないものと見なせる。
' DOS 5.0A に付属の SYMDEB.EXE にも対応.
'................................................................
Check.Stack:
  DIM SP.adr(8)
  RESTORE SP.Immediate.Adr
  FOR I = 1 TO 8
    READ D$
    SP.adr(I) = VAL("&H" + D$)
  NEXT
  SP.value = fnPeekW(SP.adr(8))
  FOR I = 2 TO 7
    IF fnPeekW(SP.adr(I)) <> SP.value THEN GOTO Err.Version
  NEXT
  IF fnPeekW(SP.adr(1)) <> SP.value THEN
    SP.adr(1) = SP.adr(1) - 1
    IF fnPeekW(SP.adr(1)) <> SP.value THEN GOTO Err.Version
    DOS.5.0A = -1
  END IF
  PRINT "  SYMDEBのバージョンに違いはないと思われます."
  RETURN
'----------------------------------------------------------------
'      スタックレジスタの初期値を変更する（大きくする）.
' スタック領域はプログラムの末尾に置かれるので，プログラムを拡大した
' ときには，スタック領域を繰り下げることが必要になる。
' スタック領域の大きさ=&H100バイト.
'................................................................
Change.Stack.Area:
  SP.value = (DataArea + &H10F) AND &HFFF0
  FOR I = 1 TO 8
    Adr = SP.adr(I)
    V = SP.value
    GOSUB PokeW
  NEXT
  PRINT "  スタック領域を後退させました."
  RETURN
'................................................................
'      スタックポインタを初期化しているところ 8箇所.
SP.Immediate.Adr:
  DATA  112, 506, 947, 958, 9DD, 2BFB, 4AA4, 58D2
'----------------------------------------------------------------
```

```
' MS-DOS Ver. 5.0 で バージョン番号取得ファンクション (AH＝30h) の
' 仕様が変更されたことに対応する処置.
'..............................................................
DOS.5:
 IF DOS.5.0A THEN RETURN
 RESTORE DOS5.Data
 GOSUB Write.Code
 PRINT "  MS-DOS 5.0 に対応しました."
 RETURN
' 変更前
'     B430            0123  mov    ah,30
'     CD21            0125  int    21
'     86E0            0127  xchg   ah,al
'     3D0002          0129  cmp    ax,200
' 変更後
DOS5.Data:
 DATA                0123
 DATA B430         : '0123  mov    ah,30
 DATA B000         : '0125  mov    al,00
 DATA CD21         : '0127  int    21
 DATA 3C02         : '0129  cmp    al,02
 DATA 90           : '012B  nop
 DATA $
'--------------------------------------------------------------
' ＤＬ，ＤＴコマンドにより表示される実数の指数部を正確にする.
'  変更前  11C5:  D0D0    RCL    AL,1
'  変更後  11C5:  D1D0    RCL    AX,1
'..............................................................
Fix.DL.DT:
 A = &H11C5
 IF PEEK(A) = &HD0 AND PEEK(A + 1) = &HD0 THEN POKE A, &HD1
 PRINT "  DL,DTコマンドのバグを修正しました."
 RETURN
'--------------------------------------------------------------
' Ａコマンドによる FSTP命令のアセンブルのバグを修正する.
'..............................................................
Fix.FSTP:
 A = &H7AA3
 IF PEEK(A) = &HB THEN POKE A, &H2B
 PRINT "  FSTP命令のアセンブルのバグを修正しました."
 RETURN
'--------------------------------------------------------------
' ＤＡ，ＤＢコマンドによるダンプで漢字表示をする (堀 英明氏による).
' CTRL＋K を入力することにより, 漢字モード←→非漢字モードの切替え可能.
'..............................................................
KNJI:
'          call 0A60 → call 8D4E
 Adr = &H54A: V = &H8802
 GOSUB PokeW
 RESTORE KANJI.Data
 GOSUB Write.Code
 GOSUB Write.Code
 GOSUB Write.Code
 PRINT "  ＤＡ，ＤＢコマンドを漢字対応にしました."
 RETURN

KANJI.Data:
 DATA                1250                        ' ＤＡコマンド
 DATA 9090           : '1250  nop: nop
 DATA 26803E706B00   : '1252  cmp    Byte ES:[6B70],00
 DATA 7504           : '1258  jne    125E
 DATA 3C00           : '125A  cmp    AL,00
 DATA 7418           : '125C  je     1276
 DATA 4E             : '125E  dec    SI
 DATA 2EC6062E8D31   : '125F  mov    Byte CS:[8D2E],31
 DATA E8547A         : '1265  call   8CBC
 DATA 909090         : '1268  nop: nop: nop
 DATA $
 DATA                1452                        ' ＤＢコマンド
 DATA 51             : '1452  push   CX
 DATA 89D9           : '1453  mov    CX,BX
 DATA 2EC6062E8D11   : '1455  mov    Byte CS:[8D2E],11
 DATA E85E78         : '145B  call   8CBC
 DATA 31DB           : '145E  xor    BX,BX
 DATA 59             : '1460  pop    CX
 DATA C3             : '1461  ret
 DATA $
                       '8CBA  db     漢字の第２バイトの先行出力フラグ
                       '8CBB  db     このサブルーチンによる出力バイト数
 DATA                8CBC
 DATA 33C0           : '8CBC  xor    AX,AX
 DATA 2E8706BA8C     : '8CBE  xchg   AX,CS:[8CBA]
 DATA 0AC0           : '8CC3  or     AL,AL
 DATA 740C           : '8CC5  jz     8CD3
 DATA B020           : '8CC7  mov    AL,' '     先頭バイトが漢字の第２バイト
 DATA E86B00         : '8CC9  call   8D37
 DATA 4B             : '8CCC  dec    BX
 DATA 7502           : '8CCD  jnz    8CD1
 DATA 43             : '8CCF  inc    BX
 DATA C3             : '8CD0  ret
 DATA 46             : '8CD1  inc    SI
 DATA 49             : '8CD2  dec    CX
 DATA AC             : '8CD3  lodsb
 DATA 2E22064D8D     : '8CD4  and    AL,CS:[8D4D]    7F(ANK) or FF(KNJI)
 DATA 3C81           : '8CD9  cmp    AL,81
 DATA 722C           : '8CDB  jb     8D09
 DATA 3CA0           : '8CDD  cmp    AL,A0
 DATA 7208           : '8CDF  jb     8CE9
 DATA 3CE0           : '8CE1  cmp    AL,E0
 DATA 7224           : '8CE3  jb     8D09
 DATA 3CFC           : '8CE5  cmp    AL,FC
 DATA 7720           : '8CE7  ja     8D09
 DATA 8AE0           : '8CE9  mov    AH,AL
 DATA AC             : '8CEB  lodsb
 DATA 3C3F           : '8CEC  cmp    AL,3F
 DATA 7216           : '8CEE  jb     8D06
 DATA 3CFD           : '8CF0  cmp    AL,FD
 DATA 7712           : '8CF2  ja     8D06
 DATA 86E0           : '8CF4  xchg   AH,AL
 DATA E83E00         : '8CF6  call   8D37
 DATA 86E0           : '8CF9  xchg   AH,AL
 DATA E83900         : '8CFB  call   8D37
 DATA 49             : '8CFE  dec    CX
 DATA 7427           : '8CFF  jz     8D28
 DATA 4B             : '8D01  dec    BX
 DATA 7423           : '8D02  jz     8D27
 DATA EB18           : '8D04  jmp    8D1E
 DATA 4E             : '8D06  dec    SI
 DATA 8AC4           : '8D07  mov    AL,AH
 DATA 3C20           : '8D09  cmp    AL,20
 DATA 720C           : '8D0B  jb     8D19
 DATA 3C7F           : '8D0D  cmp    AL,7F
 DATA 720A           : '8D0F  jb     8D1B
 DATA 3CA1           : '8D11  cmp    AL,A1
 DATA 7204           : '8D13  jb     8D19
 DATA 3CDF           : '8D15  cmp    AL,DF
 DATA 7202           : '8D17  jb     8D1B
 DATA B02E           : '8D19  mov    AL,'.'
 DATA E81900         : '8D1B  call   8D37
 DATA 49             : '8D1E  dec    CX
 DATA 7408           : '8D1F  jz     8D29
 DATA 4B             : '8D21  dec    BX
 DATA 75AF           : '8D22  jnz    8CD3
 DATA 43             : '8D24  inc    BX
 DATA EB02           : '8D25  jmp    8D29
 DATA 43             : '8D27  inc    BX
```

```
DATA 4E               : '8D28  dec    SI
DATA 2EA0BB8C         : '8D29  mov    AL,CS:[8CBB]
DATA 3C11             : '8D2D  cmp    AL,11            11 or 31
DATA 7204             : '8D2F  jb     8D35
DATA 2EA2BA8C         : '8D31  mov    CS:[8CBA],AL
DATA 41               : '8D35  inc    CX
DATA C3               : '8D36  ret
' 1バイトのデータを出力バッファに記入する.
DATA 57               : '8D37  push   DI
DATA BF6A69           : '8D38  mov    DI,696A
DATA 2E033EEA69       : '8D3B  add    DI,CS:[69EA]
DATA AA               : '8D40  stosb
DATA 2EFF06EA69       : '8D41  inc    Word CS:[69EA]
DATA 2EFE06BB8C       : '8D46  inc    Byte CS:[8CBB]
DATA 5F               : '8D4B  pop    DI
DATA C3               : '8D4C  ret

' ctrl+K の入力を検査するサブルーチン.
DATA FF               : '8D4D  db   デフォルトを非漢字モードにしたければ 7F
DATA E80F7D           : '8D4E  call   0A60   キー入力
DATA 740C             : '8D51  jz     8D5F
DATA 3C0B             : '8D53  cmp    AL,0B   ctrl+K
DATA 7508             : '8D55  jne    8D5F
DATA 2E80364D8D80     : '8D57  xor    Byte CS:[8D4D],80    トグル
DATA 32C0             : '8D5D  xor    AL,AL    「キー入力はなし」とする.
DATA C3               : '8D5F  ret
DATA $
'-----------------------------------------------------------------
' J, Y コマンドを追加する。
' J コマンド:  Near Jmp 命令と2バイト分岐命令をサーチする。
' Y コマンド:  Near Call 命令をサーチする。
'.................................................................
JANDEB:
'          コマンド処理アドレステーブルを書き換える。
'                          J コマンド: 155E → 8D9E.
 Adr = &HB0B: V = &H8D9E
 GOSUB PokeW
'                          Y コマンド: 155E → 8D63.
 Adr = &HB29: V = &H8D63
 GOSUB PokeW
'                          J, Y コマンド用の処理サブルーチンのコードを記入する.
 RESTORE JY.Code
 GOSUB Write.Code
 PRINT "  J, Yコマンドの使用を可能にしました."
 RETURN

JY.Code:
 DATA                  8D60
 DATA E9FF87         : '8D60  jmp    1562     エラー処理
 ' 8D63 = Near Call, Near Jmp 命令を検索する共通サブルーチン
 ' Default segment address = [7CB0] = CS
 DATA A1B07C         : '8D63  mov    AX,[7CB0]
 DATA E843B7         : '8D66  call   44AC
 DATA 72F5           : '8D69  jb     8D60
 DATA 51             : '8D6B  push   CX
 DATA 50             : '8D6C  push   AX
 DATA 52             : '8D6D  push   DX
 DATA E8D4B7         : '8D6E  call   4545
 DATA 5F             : '8D71  pop    DI
 DATA 07             : '8D72  pop    ES
 DATA 59             : '8D73  pop    CX
 DATA 72EA           : '8D74  jb     8D60
                       ' ES = 検索セグメントアドレス
                       ' DI = 検索開始アドレス
                       ' CX = 検索領域の長さ (バイト数)
                       ' [6A76]=Call/Jmp の行き先アドレス
 DATA FC             : '8D76  cld
 DATA 57             : '8D77  push   DI
 DATA 51             : '8D78  push   CX
 DATA E317           : '8D79  jcxz   8D92
 DATA B0E8           : '8D7B  mov    AL,E8      call(E8)/jmp(E9)
 DATA F2AE           : '8D7D  repne  scasb
 DATA 7511           : '8D7F  jne    8D92
 DATA 268B1D         : '8D81  mov    BX,ES:[DI]
 DATA 8D4102         : '8D84  lea    AX,[BX+DI+02]
 DATA 3B06766A       : '8D87  cmp    AX,[6A76]
 DATA 75EC           : '8D8B  jne    8D79
 DATA E80500         : '8D8D  call   8D95
 DATA EBE7           : '8D90  jmp    8D79
 DATA 59             : '8D92  pop    CX
 DATA 5F             : '8D93  pop    DI
 DATA C3             : '8D94  ret
' アドレスを表示するサブルーチン.
 DATA 4F             : '8D95  dec    DI
 DATA E8247D         : '8D96  call   0ABD      アドレスを出力
 DATA 47             : '8D99  inc    DI
 DATA E8787C         : '8D9A  call   0A15      改行
 DATA C3             : '8D9D  ret

' J コマンド用サブルーチン.
' 初めに Near Jmp 命令を検索する
 DATA FE067C8D       : '8D9E  inc    Byte [8D7C]  ; E8→E9
 DATA E8BEFF         : '8DA2  call   8D63
 DATA FE0E7C8D       : '8DA5  dec    Byte [8D7C]  ; E9→E8
' 次に 2バイト分岐命令を検索する。
 DATA A1766A         : '8DA9  mov    AX,[6A76]  分岐先アドレス
 DATA 8BD8           : '8DAC  mov    BX,AX
 DATA 03CF           : '8DAE  add    CX,DI
 DATA 49             : '8DB0  dec    CX
 DATA 057E00         : '8DB1  add    AX,007E
 DATA 7303           : '8DB4  jnb    8DB9
 DATA B8FEFF         : '8DB6  mov    AX,FFFE
 DATA 3BC1           : '8DB9  cmp    AX,CX
 DATA 7302           : '8DBB  jae    8DBF
 DATA 8BC8           : '8DBD  mov    CX,AX       上限アドレス
 DATA 81EB8100       : '8DBF  sub    BX,0081
 DATA 7302           : '8DC3  jnb    8DC7
 DATA 33DB           : '8DC5  xor    BX,BX
 DATA 3BDF           : '8DC7  cmp    BX,DI       下限アドレス
 DATA 7602           : '8DC9  jbe    8DCD
 DATA 8BFB           : '8DCB  mov    DI,BX
 DATA 2BCF           : '8DCD  sub    CX,DI
 DATA 722B           : '8DCF  jb     8DFC
 DATA 41             : '8DD1  inc    CX
 DATA 268A05         : '8DD2  mov    AL,ES:[DI]
 DATA 47             : '8DD5  inc    DI
 DATA 3CEB           : '8DD6  cmp    AL,EB       jmps
 DATA 7410           : '8DD8  jz     8DEA
 DATA 3CE3           : '8DDA  cmp    AL,E3       jcxz
 DATA 771C           : '8DDC  ja     8DFA        loop
 DATA 3CE0           : '8DDE  cmp    AL,E0       loopz
 DATA 7308           : '8DE0  jae    8DEA        loopnz
 DATA 3C70           : '8DE2  cmp    AL,70
 DATA 7214           : '8DE4  jb     8DFA
 DATA 3C7F           : '8DE6  cmp    AL,7F       jo(70)～jg(7F)
 DATA 7710           : '8DE8  ja     8DFA
 DATA 268A05         : '8DEA  mov    AL,ES:[DI]
 DATA 98             : '8DED  cbw
 DATA 03C7           : '8DEE  add    AX,DI
 DATA 40             : '8DF0  inc    AX
 DATA 3B06766A       : '8DF1  cmp    AX,[6A76]
 DATA 7503           : '8DF5  jne    8DFA
 DATA E89BFF         : '8DF7  call   8D95
 DATA E2D6           : '8DFA  loop   8DD2
 DATA C3             : '8DFC  ret
 DATA $
```

```
'------------------------------------------------------------------
'      ブレークポイントでの Opcode 復元動作を変更する.
'..................................................................
Change.BP:
 RESTORE BP.Code
  GOSUB Write.Code
  GOSUB Write.Code
  GOSUB Write.Code
 PRINT "  ブレークポイントの動作を変更しました."
 RETURN

BP.Code:
'     変 更 前
'                 26EA   mov    AL,[SI+4]     ; Gコマンドのブレークポイント
'                        stosb                ; Opcode を復元する

'                 2701   mov    AL,[SI+9]     ; BPコマンドのブレークポイント
'                        stosb                ; Opcode を復元する
'     変 更 後
 DATA                 26EA
 DATA E81367     : '26EA  call   8E00         ; エリア不足のため
 DATA 90         : '      nop                 ; サブルーチンにより処理する.
 DATA $

 DATA                 2701
 DATA E80767     : '2701  call   8E0B         ; エリア不足のため
 DATA 90         : '      nop                 ; サブルーチンにより処理する.
 DATA $

 DATA                 8E00
 DATA 26803DCC : '8E00  cmp    byte ES:[DI],CC
 DATA 7504     : '8E04  jne    8E0A
 DATA 8A4404   : '8E06  mov    AL,[SI+4]
 DATA AA       : '8E09  stosb                  ; CC のときに限り復元する.
 DATA C3       : '8E0A  ret

 DATA 268A25   : '8E0B  mov    AH,ES:[DI]      ; BPの現在の1バイト.
 DATA 8A4409   : '8E0E  mov    AL,[SI+09]      ; BPの本来の1バイト.
 DATA 80FCCC   : '8E11  cmp    AH,CC           ; 0CCh のときに限り復元する.
 DATA 74F3     : '8E14  jz     8E09
 DATA 3AE0     : '8E16  cmp    AH,AL           ; 書き換えられたか?
 DATA 7403     : '8E18  jz     8E1D
 DATA 8024FD   : '8E1A  and    byte [SI],FD    ; disabled に変更する.
 DATA C3       : '8E1D  ret
 DATA $
'------------------------------------------------------------------
'  コプロセッサ命令  FSTSW  AX  をアセンブル/逆アセンブル可能にする.
'   FSTSW AX = DFE0
'                E0 = 11 100 000 (第2 Opcode)
'                     mod =  11b = 3
'                     reg = 100b = 4
'                     r/m = 000b = 0
'..................................................................
Enable.FSTSW.AX:
 Adr = &H7A56: V = &H8E20: GOSUB PokeW
 Adr = &H4085: V = &H4DC3: GOSUB PokeW
 RESTORE FSTSW.Code
 GOSUB Write.Code
 GOSUB Write.Code
 PRINT "  コプロセッサ命令 FSTSW AX の使用を可能にしました."
 RETURN

FSTSW.Code:
'      変更前
' 7A56:          dw     31C7          ; アセンブル用テーブル中のアドレス
' 7436:          dw     4084          ; 逆アセンブル用テーブル中のアドレス
' 4084: E8A901   call   4230          ; Opcode DF の処理サブルーチン

'      変更後
' 7A56:          dw     8E20          ; ニモニック "FSTSW" の処理アドレス
' 7436:          dw     4084          ; Opcode DF の処理アドレス (変更せず)
' 4084: E8C34D   call   8E4A          ; Opcode DF の処理サブルーチン

 DATA                8E20                   ' FSTSW AX をアセンブル可能にする.
 DATA 8B0E5B6B       : '8E20  mov   CX,[6B5B]     ; 読み出しアドレスを保存.
 DATA E8397C         : '8E24  call  A60           ; 1文字を読む.
 DATA 7418           : '8E27  je    8E41
 DATA 8AE0           : '8E29  mov   AH,AL
 DATA E8327C         : '8E2B  call  A60           ; 次の1文字を読む.
 DATA 3D5841         : '8E2E  cmp   AX,"AX"       ; "AX" ?
 DATA 750E           : '8E31  jne   8E41
 DATA E82A7C         : '8E33  call  A60
 DATA 7509           : '8E36  jne   8E41          ; 行末検査
 DATA C706EA6DDFE0   : '8E38  mov word [6DEA],E0DF   ; 命令コード DFE0
 DATA E970A0         : '8E3E  jmp   2EB1
 DATA 890E5B6B       : '8E41  mov   [6B5B],CX     ; 読み出しアドレスを復元.
 DATA 8A07           : '8E45  mov   AL,[BX]
 DATA E97DA3         : '8E47  jmp   31C7          ; 本来の処理ルーチンへ
 DATA $

 DATA                8E4A                   ' DFE0 を逆アセンブル可能にする.
 DATA E8E3B3         : '8E4A  call  4230          ; 第2 Opcode を読む.
 DATA 9F             : '8E4D  lahf                ; フラグレジスタを退避
 DATA 7520           : '8E4E  jne   8E70          ; mod≠3
 DATA 2407           : '8E50  and   AL,7
 DATA 3C04           : '8E52  cmp   AL,4
 DATA 751A           : '8E54  jne   8E70          ; reg≠4
 DATA 803E946D00     : '8E56  cmp   byte [6D94],0
 DATA 7513           : '8E5B  jne   8E70          ; r/m≠0
 DATA B85354         : '8E5D  mov   AX,'ST'
 DATA AB             : '8E60  stosw
 DATA B85357         : '8E61  mov   AX,'SW'
 DATA AB             : '8E64  stosw
 DATA B82020         : '8E65  mov   AX,'  '
 DATA AB             : '8E68  stosw
 DATA B84158         : '8E69  mov   AX,'AX'
 DATA AB             : '8E6C  stosw
 DATA 44             : '8E6D  inc   SP
 DATA 44             : '8E6E  inc   SP
 DATA C3             : '8E6F  ret                 ; 処理を終了.
 DATA 9E             : '8E70  sahf                ; フラグレジスタを復元
 DATA C3             : '8E71  ret                 ; 本来の処理ルーチンへ
 DATA $
'------------------------------------------------------------------
' SYMDEB.EXE では, 起動時にスイッチとして /S をつけることにより
' 画面を切り替えられるようになっている。この機能を IBM-PC だけでなく
' PC-98 でも使えるように改造する。
' ノーマルモード 20/25行, ハイレゾモード 25/31行をすべてサポートした。
' 「30行計画」にも (行間空きモードを含めて) 対応。
'..................................................................
Enable.Screen.Flip:
 RESTORE Flip.Code
 GOSUB Write.Code
 GOSUB Write.Code
 GOSUB Write.Code
 GOSUB Write.Code
 GOSUB Write.Code
 GOSUB Write.Code
 PRINT "  PC-98 でも, スイッチ /S により画面切り替え機能が使えます."
 PRINT "    この機能は, 「30行計画」にも対応しています."
 RETURN

Flip.Code:
'        変 更 前
'                    02E6  cmp   byte [6B65],FF  ; /S指定 フラグ
'                    02EB  je    02F4            ; /S指定なしのとき
```

```
'                     58E9  call   5A25

'                     5918  call   5A25

'                     5937  call   5A25

'           変 更 後
DATA                   02EB
DATA 9090          : '02EB  nop: nop      ; je 命令を無効にする
DATA $
DATA                   58E9
DATA E8C435        : '58E9  call   8EB0    ; サブルーチン呼び出しを変更する
DATA $
DATA                   5918
DATA E8DC35        : '5918  call   8EF7
DATA $
DATA                   5937
DATA E8F535        : '5937  call   8F2F
DATA $

DATA                   8E90
DATA 1B5B3E336824  : '8E90  db   1Bh,"[>3h$"  20行モード
DATA 1B5B3E336C24  : '8E96  db   1Bh,"[>3l$"  25行モード
DATA 1B5B3E336E24  : '8E9C  db   1Bh,"[>3n$"  31行モード
DATA 00A0          : '8EA2  dw     テキスト用VRAM セグメントアドレス
DATA 00            : '8EA4  db      0/1  (Normal/High resolution)
'                     8EA5  db      0/ 1/ 2 行モード データ   MS-DOS
'                     8EA6  db    ( 20/25/31 行モード )       SYMDEB
'                     8EA7  db      行数 MS-DOS
'                     8EA8  db      行数 SYMDEB
DATA $

' 初期化: テキストVRAMのセグメントアドレスを決める.
' SYMDEB 起動時の MS-DOS画面を保存して, SYMDEB用画面をクリアする.
DATA                   8EB0
DATA 803E616BFF    : '8EB0  cmp    byte [6B61],FF ; IBM flag
DATA 7503          : '8EB5  jne    8EBA
DATA E96BCB        : '8EB7  jmp    5A25           ; IBM on
DATA 1E            : '8EBA  push   DS             ; IBM off
DATA E8EF00        : '8EBB  call   8FAD
DATA F606010508    : '8EBE  test   byte [0501],08 ; ハイレゾ判定
DATA 1F            : '8EC3  pop    DS
DATA 740D          : '8EC4  jz     8ED3
DATA FEC0          : '8EC6  inc    AL             ; 1/2
DATA C606A48E01    : '8EC8  mov    byte [8EA4],01 ; High resolution
DATA C706A28E00E0  : '8ECD  mov    word [8EA2],E000  ; TVRAM segment
DATA 8916CF6A      : '8ED3  mov    [6ACF],DX  ; カーソル位置を保存.
DATA A2A58E        : '8ED7  mov    [8EA5],AL  ; 行モードを保存.
DATA A2A68E        : '8EDA  mov    [8EA6],AL
DATA 881EA78E      : '8EDD  mov    [8EA7],BL  ; 行数を保存.
DATA 881EA88E      : '8EE1  mov    [8EA8],BL
DATA E87F00        : '8EE5  call   8F67       ; MS-DOSテキスト画面を保存
DATA B21A          : '8EE8  mov    DL,1A
DATA B400          : '8EEA  mov    AH,00      ; SYMDEB用の
DATA E8F800        : '8EEC  call   8FE7       ; テキスト画面を消去する.
DATA C606D96AFF    : '8EEF  mov    byte [6AD9],FF  ; swap toggle
DATA 44            : '8EF4  inc    SP
DATA 44            : '8EF5  inc    SP
DATA C3            : '8EF6  ret               ; もとの呼び出しルーチンへ.

' プログラムからの出力画面を保存して SYMDEB の画面に切り替える.
' 画面切り替え時には, 行モードとカーソル位置も切り替える.
DATA 803E616BFF    : '8EF7  cmp    byte [6B61],FF
DATA 7503          : '8EFC  jne    8F01
DATA E924CB        : '8EFE  jmp    5A25       ; IBM machine.
DATA E8C600        : '8F01  call   8FCA       ; 画面データを取得.
DATA A2A58E        : '8F04  mov    [8EA5],AL  ; 行モードを保存
DATA 881EA78E      : '8F07  mov    [8EA7],BL  ; 行数を保存.
DATA 8916CF6A      : '8F0B  mov    [6ACF],DX  ; カーソル位置を保存.
DATA E85500        : '8F0F  call   8F67       ; MS-DOS画面を保存する.
DATA A0A68E        : '8F12  mov    AL,[8EA6]  ; SYMDEB画面の行モード.
DATA 3A06A58E      : '8F15  cmp    AL,[8EA5]  ; 現在の行モード.
DATA 7403          : '8F19  jz     8F1E       ; 等しくなければ,
DATA E8B600        : '8F1B  call   8FD4       ;   行モードを変更する.
DATA 8A1EA88E      : '8F1E  mov    BL,[8EA8]  ; SYMDEB画面の行数.
DATA E86900        : '8F22  call   8F8E       ; SYMDEB画面をロードする.
DATA 8B16D36A      : '8F25  mov    DX,[6AD3]  ; 新しいカーソル位置を
DATA E8B900        : '8F29  call   8FE5       ;   設定する.
DATA 44            : '8F2C  inc    SP
DATA 44            : '8F2D  inc    SP
DATA C3            : '8F2E  ret               ; もとの呼び出しルーチンへ.

' SYMDEB の出力画面を保存して, プログラムの出力画面に切り替える.
' 画面切り替え時には, 行モードとカーソル位置も切り替える.
DATA 803E616BFF    : '8F2F  cmp    byte [6B61],FF
DATA 7503          : '8F34  jne    8F39
DATA E9ECCA        : '8F36  jmp    5A25       ; IBM machine.
DATA E88E00        : '8F39  call   8FCA       ; 画面データを取得.
DATA A2A68E        : '8F3C  mov    [8EA6],AL  ; 行モードを保存.
DATA 881EA88E      : '8F3F  mov    [8EA8],BL  ; 行数を保存.
DATA 8916D36A      : '8F43  mov    [6AD3],DX  ; カーソル位置を保存
DATA E82300        : '8F47  call   8F6D       ; SYMDEB画面を保存する.
DATA A0A58E        : '8F4A  mov    AL,[8EA5]  ; MS-DOS画面の行モード.
DATA 3A06A68E      : '8F4D  cmp    AL,[8EA6]  ; 現在の行モード.
DATA 7403          : '8F51  jz     8F56       ; 等しくなければ,
DATA E87E00        : '8F53  call   8FD4       ;   行モードを変更する.
DATA 8A1EA78E      : '8F56  mov    BL,[8EA7]  ; MS-DOS画面の行数.
DATA E82B00        : '8F5A  call   8F88       ; MS-DOS画面をロードする.
DATA 8B16CF6A      : '8F5D  mov    DX,[6ACF]  ; 新しいカーソル位置を
DATA E88100        : '8F61  call   8FE5       ;   設定する.
DATA 44            : '8F64  inc    SP
DATA 44            : '8F65  inc    SP
DATA C3            : '8F66  ret               ; もとの呼び出しルーチンへ.

DATA 8E06DC6A      : '8F67  mov    ES,[6ADC]  ; MS-DOS画面を保存する.
DATA EB04          : '8F6B  jmp    8F71
DATA 8E06DA6A      : '8F6D  mov    ES,[6ADA]  ; SYMDEB画面を保存する.
DATA 8E1EA28E      : '8F71  mov    DS,[8EA2]  ; TVRAM セグメントアドレス
DATA E82A00        : '8F75  call   8FA2       ; 転送ワード数を得る.
DATA 51            : '8F78  push   CX
DATA F3A5          : '8F79  rep    movsw      ; 画面データを転送する.
DATA BE0020        : '8F7B  mov    SI,2000
DATA 59            : '8F7E  pop    CX
DATA F3A5          : '8F7F  rep    movsw      ; 属性データを転送する.
DATA 8CC8          : '8F81  mov    AX,CS      ; セグメントレジスタを
DATA 8ED8          : '8F83  mov    DS,AX      ;   復元する.
DATA 8EC0          : '8F85  mov    ES,AX
DATA C3            : '8F87  ret

DATA 8E1EDC6A      : '8F88  mov    DS,[6ADC]  ; MS-DOS画面を表示する.
DATA EB04          : '8F8C  jmp    8F92
DATA 8E1EDA6A      : '8F8E  mov    DS,[6ADA]  ; SYMDEB画面を表示する.
DATA 2E8E06A28E    : '8F92  mov    ES,CS:[8EA2] ; TVRAM セグメントアドレス
DATA E80800        : '8F97  call   8FA2       ; 転送ワード数を得る.
DATA 51            : '8F9A  push   CX
DATA F3A5          : '8F9B  rep    movsw
DATA BF0020        : '8F9D  mov    DI,2000
DATA EBDC          : '8FA0  jmp    8F7E

DATA B050          : '8FA2  mov    AL,50      ; 80 を
DATA F6E3          : '8FA4  mul    BL         ; 行数にかけて
DATA 8BC8          : '8FA6  mov    CX,AX      ; 転送ワード数を得る.
DATA 33F6          : '8FA8  xor    SI,SI      ; 転送元オフセットアドレス
DATA 33FF          : '8FAA  xor    DI,DI      ; 転送先オフセットアドレス
DATA C3            : '8FAC  ret
```

```
 DATA 33C0           : '8FAD  xor   AX,AX      ; MS-DOSワークエリアを読む.
 DATA 8ED8           : '8FAF  mov   DS,AX      ; DS = 0000
 DATA 8A361007       : '8FB1  mov   DH,[0710]  ; カーソルY座標
 DATA 8A161C07       : '8FB5  mov   DL,[071C]  ; カーソルX座標
 DATA A01307         : '8FB9  mov   AL,[0713]  ; 行モードパラメタ 0/1
 DATA 8A1E1207       : '8FBC  mov   BL,[0712]  ; 行数－1
 DATA 803E110701     : '8FC0  cmp   byte [0711],01
 DATA 80DBFF         : '8FC5  sbb   BL,FF
 DATA 43             : '8FC8  inc   BX         ; BL = 行数
 DATA C3             : '8FC9  ret

 DATA 1E             : '8FCA  push  DS
 DATA E8DFFF         : '8FCB  call  8FAD
 DATA 1F             : '8FCE  pop   DS
 DATA 0206A48E       : '8FCF  add   AL,[8EA4]  ; ハイレゾのとき ＋1
 DATA C3             : '8FD3  ret

 DATA 50             : '8FD4  push  AX
 DATA B406           : '8FD5  mov   AH,06
 DATA F6E4           : '8FD7  mul   AH
 DATA BA908E         : '8FD9  mov   DX,8E90    ; 行モード設定データ
 DATA 03D0           : '8FDC  add   DX,AX
 DATA B401           : '8FDE  mov   AH,01
 DATA E80400         : '8FE0  call  8FE7       ; 行モードを設定する.
 DATA 58             : '8FE3  pop   AX
 DATA C3             : '8FE4  ret

 DATA B403           : '8FE5  mov   AH,03      ; カーソル位置を設定する.
 DATA B110           : '8FE7  mov   CL,10
 DATA CDDC           : '8FE9  int   DC
 DATA C3             : '8FEB  ret
 DATA $
'--------------------------------------------------------------------
' EXEPACK されたファイルもデバッグ可能にする.
'....................................................................
Packed.Files:
 RESTORE Packed.Code
 GOSUB Write.Code
 Adr = &H49D
 M$ = "Warning: packed file!    "
 GOSUB Poke.Str
 GOSUB Write.Code
 M$ = ";>NUL;G" + ";T;G" + ";T;>CON;R" + CHR$(13)
 GOSUB Poke.Str
 PRINT "  EXEPACKされたファイルもデバッグ可能です."
 RETURN
Packed.Code:
'          変 更 前
'                    0497  jmp    0B9A
'                    049D  db     "Can't debug packed files!"
'          変 更 後
 DATA                0497
 DATA E9568B       : '0497  jmp    8FF0
 DATA $

 DATA                8FF0
 DATA B90002       : '8FF0  mov    CX,200
 DATA B050         : '8FF3  mov    AL,50            ; push  AX
 DATA F2AE         : '8FF5  repne  scasb
 DATA 7535         : '8FF7  jne    902E
 DATA 26803DCB     : '8FF9  cmp    byte ES:[DI],CB  ; retf
 DATA 75F6         : '8FFD  jne    8FF5
 DATA 8BEF         : '8FFF  mov    BP,DI
 DATA B02E         : '9001  mov    AL,2E            ; CS:
 DATA F2AE         : '9003  repne  scasb
 DATA 7527         : '9005  jne    902E
 DATA 26813DFF2F   : '9007  cmp    word ES:[DI],2FFF ; jmp far [BX]
 DATA 75F5         : '900C  jne    9003
 DATA 4F           : '900E  dec    DI
 DATA 57           : '900F  push   DI
 DATA BE3190       : '9010  mov    SI,9031
 DATA 0E           : '9013  push   CS
 DATA 07           : '9014  pop    ES
 DATA BF1A69       : '9015  mov    DI,691A        バッファ領域
 DATA B90700       : '9018  mov    CX,7
 DATA F3A4         : '901B  rep    movsb          ";>NUL;G"
 DATA 8BD5         : '901D  mov    DX,BP
 DATA E897AD       : '901F  call   3DB9           DX を 16進数出力
 DATA B104         : '9022  mov    CL,4
 DATA F3A4         : '9024  rep    movsb          ";T;G"
 DATA 5A           : '9026  pop    DX
 DATA E88FAD       : '9027  call   3DB9
 DATA B10A         : '902A  mov    CL,0A
 DATA F3A4         : '902C  rep    movsb          ";T;>CON;R",0D
 DATA E98874       : '902E  jmp    04B9
 DATA $
'--------------------------------------------------------------
' ８０８７のレジスタ内容をダンプ表示する７コマンドを新設する。
'..............................................................
Dump.Reg87:
 Adr = &H904C: V = -1: GOSUB PokeW
 Adr = &H90AE: V = &H3FF: GOSUB PokeW
 Adr = &H90B0: V = &H7FF: GOSUB PokeW
 Adr = &H90D2: M$ = "unnormal": GOSUB Poke.Str
               M$ = "denormal": GOSUB Poke.Str
               M$ = "empty   ": GOSUB Poke.Str
               M$ = "infinity": GOSUB Poke.Str
               M$ = "NaN     ": GOSUB Poke.Str
 Adr = &H90FA: V = 10: GOSUB PokeW
 Adr = &H90FC + 8: POKE Adr, &H10
 Adr = &H9106: M$ = "ST(0) =" + SPACE$(33) + "$": GOSUB Poke.Str
 RESTORE Dump.Reg87.Code
 GOSUB Write.Code
 GOSUB Write.Code
 GOSUB Write.Code
 PRINT "  8087用 7コマンドの使用を可能にしました."
 RETURN

' 904C    dw               8087使用可能フラグ(ゼロのとき使用可).
' 904E    dw      ?        8087 status word
'    FSAVE のためのエリア, 94bytes.
' 9050    dw               8087 control word
' 9052    dw               8087 status word
' 9054    dw               Tag word
' 9056    dd               Instruction pointer(参照していない)
' 905A    dd               Data pointer       (参照していない)
' 905E    dt      8 dup(?) 8個のスタックレジスタ内容

' 90AE    dw      03FF     Round to nearest
' 90B0    dw      07FF     Round down
' 90B2    db      8 dup(?) タグの値 8個
' 90BA    dw               指数
' 90BC    dt               指数部(packed)
' 90C6    db               指数部の符号
' 90C7    db               符号
' 90C8    dt               仮数部(packed)
' 90D2    db      'unnormal'
' 90DA    db      'denormal'
' 90E2    db      'empty   '
' 90EA    db      'infinity'
' 90F2    db      'NaN     '
' 90FA    dw      0A       整数 10
' 90FC    dt      00100000000000000000h   =  1.0e17
' 9106    db      'ST(0) = ', 32 dup (' '), '$'    出力用バッファ

```

```
950: Dump.Reg87.Code:
951: ' 変 更 前
952: '               0263  jmp    02BF
953: '               05A9  call   4347
954: ' 変 更 後
955:  DATA           0263
956:  DATA E9CA8E   : '0263  jmp    9130
957:  DATA $
958:  DATA           05A9
959:  DATA E8938B   : '05A9  call   913F
960:  DATA $
961:
962: ' 8087 が使用できるかどうかを調べる.
963:  DATA           9130
964:  DATA 3C37     : '9130  cmp    AL,'7'       起動スイッチ /7 ?
965:  DATA 7508     : '9132  jne    913C
966:  DATA DBE3     : '9134  FNINIT              initialize 8087
967:  DATA EB00     : '9136  jmp    9138
968:  DATA DD3E4C90 : '9138  FNSTSW  [904C]      get 8087 status
969:  DATA E98071   : '913C  jmp    02BF
970:
971: ' 8087 のスタックレジスタの内容をダンプ表示する.
972:  DATA 0441     : '913F  add    AL,'A'
973:  DATA 3C37     : '9141  cmp    AL,'7'       7コマンド
974:  DATA 7404     : '9143  jz     9149
975:  DATA E8FFB1   : '9145  call   4347
976:  DATA C3       : '9148  ret
977:
978:  DATA F6064C903F  : '9149  test   byte [904C],3F
979:  DATA 75F5        : '914E  jnz    9145         8087が使えない.
980:  DATA 9BDD365090  : '9150  FSAVE  [9050]       全データをセーブする.
981:  DATA 9B          : '9155  WAIT
982:  DATA BFB290      : '9156  mov    DI,90B2      Tag byte area
983:  DATA A15490      : '9159  mov    AX,[9054]    Tag word
984:  DATA B90800      : '915C  mov    CX,0008
985:  DATA AA          : '915F  stosb
986:  DATA D1C8        : '9160  ror    AX,1
987:  DATA D1C8        : '9162  ror    AX,1
988:  DATA E2F9        : '9164  loop   915F
989:  DATA A05390      : '9166  mov    AL,[9053]
990:  DATA D0E8        : '9169  shr    AL,1
991:  DATA D0E8        : '916B  shr    AL,1
992:  DATA D0E8        : '916D  shr    AL,1
993:  DATA 8BF0        : '916F  mov    SI,AX        SI=stack top pointer
994:  DATA B90800      : '9171  mov    CX,0008
995:  DATA 51          : '9174  push   CX           8個のレジスタについて
996:  DATA 56          : '9175  push   SI             反復処理を行う.
997:  DATA 9BDBE3      : '9176  FINIT
998:  DATA BB0800      : '9179  mov    BX,0008      レジスタの順序は,
999:  DATA 2BD9        : '917C  sub    BX,CX          見やすいように
1000: DATA D1EB        : '917E  shr    BX,1         0,4,1,5,2,6,3,7
1001: DATA 7303        : '9180  jnb    9185         とする.
1002: DATA 80CB04      : '9182  or     BL,04        BX=レジスタ番号
1003: DATA 03F3        : '9185  add    SI,BX
1004: DATA 83E607      : '9187  and    SI,+07
1005: DATA 8AB4B290    : '918A  mov    DH,[SI+90B2]  Tag data 3bits.
1006: DATA BF0E91      : '918E  mov    DI,910E      出力バッファ領域
1007: DATA 8AC3        : '9191  mov    AL,BL
1008: DATA 0430        : '9193  add    AL,'0'
1009: DATA A20991      : '9195  mov    [9109],AL    レジスタ番号
1010: DATA D1E3        : '9198  sal    BX,1
1011: DATA 8BF3        : '919A  mov    SI,BX
1012: DATA D1E3        : '919C  sal    BX,1
1013: DATA D1E3        : '919E  sal    BX,1
1014: DATA 03DE        : '91A0  add    BX,SI        BX = BX * 10
1015: DATA F6C603      : '91A2  test   DH,11b
1016: DATA 7514        : '91A5  jnz    91BB
1017: DATA E87700      : '91A7  call   9221         Valid data.
1018: DATA BA0691      : '91AA  mov    DX,9106      画面に表示する.
1019: DATA E82D7C      : '91AD  call   0DDD
1020: DATA 5E          : '91B0  pop    SI
1021: DATA 59          : '91B1  pop    CX
1022: DATA E2C0        : '91B2  loop   9174
1023: DATA 9BDD265090  : '91B4  FRSTOR [9050]       全データを復元する.
1024: DATA C390        : '91B9  ret
1025:
1026: DATA F6C602      : '91BB  test   DH,10b
1027: DATA 7513        : '91BE  jnz    91D3
1028: DATA B82030      : '91C0  mov    AX,'0 '      Tag value = 01b
1029: DATA AB          : '91C3  stosw
1030: DATA B82E30      : '91C4  mov    AX,'0.'      Zero (true zero)
1031: DATA AB          : '91C7  stosw
1032: DATA B92E91      : '91C8  mov    CX,912E      空白を埋める
1033: DATA 2BCF        : '91CB  sub    CX,DI
1034: DATA B020        : '91CD  mov    AL,' '
1035: DATA F3AA        : '91CF  rep    stosb
1036: DATA EBD7        : '91D1  jmp    91AA
1037: DATA F6C601      : '91D3  test   DH,01
1038: DATA 7405        : '91D6  jz     91DD
1039: DATA BEE290      : '91D8  mov    SI,90E2      Empty ( 11b )
1040: DATA EB3D        : '91DB  jmp    921A
1041:                                          ' Special ( 10b )
1042: DATA 9BDBAF5E90  : '91DD  FLD    tbyte [BX+905E]
1043: DATA 9BD9E5      : '91E2  FXAM
1044: DATA 9BDD3E4E90  : '91E5  FSTSW  [904E]
1045: DATA 9B          : '91EA  WAIT
1046: DATA 8A264F90    : '91EB  mov    AH,[904F]
1047: DATA B02B        : '91EF  mov    AL,'+'
1048: DATA F6C402      : '91F1  test   AH,02        C1 bit: sign
1049: DATA 7402        : '91F4  jz     91F8
1050: DATA B02D        : '91F6  mov    AL,'-'
1051: DATA AA          : '91F8  stosb
1052: DATA F6C440      : '91F9  test   AH,40        C3 bit: denormal
1053: DATA 750F        : '91FC  jnz    920D
1054: DATA F6C404      : '91FE  test   AH,04        C2 bit: infinite
1055: DATA 750F        : '9201  jnz    9212
1056: DATA F6C401      : '9203  test   AH,01        C0 bit: NaN
1057: DATA 750F        : '9206  jnz    9217
1058: DATA BED290      : '9208  mov    SI,90D2      else unnormal
1059: DATA EB0D        : '920B  jmp    921A
1060: DATA BEDA90      : '920D  mov    SI,90DA
1061: DATA EB08        : '9210  jmp    921A
1062: DATA BEEA90      : '9212  mov    SI,90EA
1063: DATA EB03        : '9215  jmp    921A
1064: DATA BEF290      : '9217  mov    SI,90F2
1065: DATA B90800      : '921A  mov    CX,0008
1066: DATA F3A4        : '921D  rep    movsb
1067: DATA EBA7        : '921F  jmp    91C8
1068:
1069: DATA 9BDBAF5E90  : '9221  FLD    tbyte [BX+905E]
1070: DATA 9BD9EC      : '9226  FLDLG2
1071: DATA 9BD9C1      : '9229  FLD    ST(1)
1072: DATA 9BD9E4      : '922C  FTST                符号を調べる.
1073: DATA 9BDD3E4E90  : '922F  FSTSW  [904E]
1074: DATA 9BD9E1      : '9234  FABS
1075: DATA B02B        : '9237  mov    AL,'+'
1076: DATA F6064F9001  : '9239  test   byte [904F],01
1077: DATA 7402        : '923E  jz     9242
1078: DATA B02D        : '9240  mov    AL,'-'
1079: DATA A2C790      : '9242  mov    [90C7],AL    符号
1080: DATA 9BD9F1      : '9245  FYL2X               10を底とするLOG(reg)
1081: DATA 9BD92EB090  : '9248  FLDCW  [90B0]       小数部切捨てモード
1082: DATA 9BD9FC      : '924D  FRNDINT             round down
1083: DATA 9BD92EAE90  : '9250  FLDCW  [90AE]       モード復元
1084: DATA 9BDF16BA90  : '9255  FIST   word [90BA]  指数
1085: DATA 9BD9E9      : '925A  FLDL2T              10^指数を計算する.
```

```
DATA 9BDEC9        : '925D  FMUL
DATA 9BD9C0        : '9260  FLD    ST(0)
DATA 9BD9FC        : '9263  FRNDINT
DATA 9BDCE9        : '9266  FSUB   ST(1),ST
DATA 9BD9C9        : '9269  FXCH                -0.5 <= ST <= 0.5
DATA 9BD9E4        : '926C  FTST                指数関数の計算
DATA 9BDD3E4E90    : '926F  FSTSW  [904E]
DATA 9BD9E1        : '9274  FABS
DATA F6064F9001    : '9277  test   byte [904F],01
DATA 9BD9F0        : '927C  F2XM1
DATA 9BD9E8        : '927F  FLD1
DATA 9BDEC1        : '9282  FADD
DATA 7406          : '9285  jz     928D
DATA 9BD9E8        : '9287  FLD1
DATA 9BDEF1        : '928A  FDIVR
DATA 9BD9FD        : '928D  FSCALE              ST(0) = 10^指数
DATA 9BDDD9        : '9290  FSTP   ST(1)        ST(1) を捨てる.
DATA 9BDEF9        : '9293  FDIV                仮数部（1未満）
DATA 9BDF26FC90    : '9296  FBLD   tbyte [90FC]
DATA 9BDCC9        : '929B  FMUL   ST(1),ST     ×1.0e17
DATA 9BDE0EFA90    : '929E  FIMUL  word [90FA]  ST(0) = 1.0e18
DATA 9BD9C9        : '92A3  FXCH
DATA 9BD9FC        : '92A6  FRNDINT
DATA 9BD8D1        : '92A9  FCOM
DATA 9BDD3E4E90    : '92AC  FSTSW  [904E]
DATA 9B            : '92B1  WAIT
DATA F6064F9001    : '92B2  test   byte [904F],01
DATA 7509          : '92B7  jnz    92C2
DATA 9BDE36FA90    : '92B9  FIDIV  word [90FA]   avoid overflow
DATA FF06BA90      : '92BE  inc    word [90BA]   ++指数
DATA 9BDF36C890    : '92C2  FBSTP  tbyte [90C8]  仮数(packed)
DATA 9BDF06BA90    : '92C7  FILD   word [90BA]
DATA 9BDF36BC90    : '92CC  FBSTP  tbyte [90BC]  指数(packed)
DATA 9B            : '92D1  WAIT
DATA B02B          : '92D2  mov    AL,'+'
DATA F606C59080    : '92D4  test   byte [90C5],80
DATA 7402          : '92D9  jz     92DD
DATA B02D          : '92DB  mov    AL,2D
DATA A2C690        : '92DD  mov    [90C6],AL     指数の符号
DATA BED090        : '92E0  mov    SI,90D0
DATA BA0800        : '92E3  mov    DX,0008
DATA A0C790        : '92E6  mov    AL,[90C7]
DATA AA            : '92E9  stosb
DATA B104          : '92EA  mov    CL,04
DATA E81D00        : '92EC  call   930C
DATA AA            : '92EF  stosb
DATA B02E          : '92F0  mov    AL,'.'
DATA AB            : '92F2  stosw
DATA E80E00        : '92F3  call   9304
DATA 47            : '92F6  inc    DI
DATA B045          : '92F7  mov    AL,'E'
DATA 8A26C690      : '92F9  mov    AH,[90C6]
DATA AB            : '92FD  stosw
DATA BEBD90        : '92FE  mov    SI,90BD
DATA BA0200        : '9301  mov    DX,0002

DATA E80500        : '9304  call   930C
DATA AB            : '9307  stosw
DATA 4A            : '9308  dec    DX
DATA 75F9          : '9309  jnz    9304
DATA C3            : '930B  ret
' packed decimal number. → decimal number.
DATA 8A24          : '930C  mov    AH,[SI]
DATA 32C0          : '930E  xor    AL,AL
DATA D3C0          : '9310  rol    AX,CL
DATA D2EC          : '9312  shr    AH,CL
DATA 053030        : '9314  add    AX,'00'
DATA 4E            : '9317  dec    SI
DATA C3            : '9318  ret
DATA $
'-------------------------------------------------------------
' INT 21h ファンクションコールへのコメントを非公開ファンションにも
' 拡張する.
' コメントは, AL によるサブファンクションについても設定できる。
'.............................................................
Int21.Comments:

 DEF fnADRptr (AH)
   IF AH < &H58 THEN
     fnADRptr = TA0 + AH * 2
   ELSE
     fnADRptr = TA + (AH - &H58) * 2
   END IF
 END DEF

 RESTORE Int21.Comment.Code
 GOSUB Write.Code
 GOSUB Write.Code

 TA0 = &H86D6                          ' アドレステーブル     00-57h
 FOR I = 0 TO &H57                     ' この部分が変更されていると
   Adr = fnPeekW(I * 2 + TA0)          ' 改造作業を中止する.
   IF Adr >= &H8CBA THEN
     PRINT
     PRINT "このファイルはオリジナルの SYMDEB ではありません."
     PRINT "改造作業を中止します."
     BEEP: END
   END IF
 NEXT

 TA = &H9350                           ' 拡張アドレステーブル 58-7Fh
 Null.Ptr = TA + &H28 * 2
 POKE Null.Ptr, 0                      '「コメントなし」を意味する.
 DataArea = Null.Ptr + 1

 V = Null.Ptr
 FOR AH = &H58 TO &H7F
   Adr = fnADRptr(AH)
   GOSUB PokeW                         ' すべて Null を指す.
 NEXT

 PRINT "  INT 21h ファンクションへのコメントを読み取り中"
 AH.old = -1
 DIM branch.table(31)
 RESTORE Comment.Data

 DO
NextComment:
   READ A$
   L = LEN(A$)
   IF (L <> 2) AND (L <> 4) THEN
     PRINT "ファンクション番号 "; A$; " はエラーです"
     BEEP: END
   END IF
   AH = VAL("&H" + LEFT$(A$, 2))           ' ファンクション番号
   IF AH >= &H80 THEN EXIT DO              ' 終了
   branch = (L = 4)
   IF branch THEN                          ' AL の枝番号つきの場合.
     AL = VAL("&H" + RIGHT$(A$, 2))
     IF AL >= 32 THEN                      ' AL>=20h のときは拒否される.
       PRINT A$; ": この枝番号は大きすぎます"
       BEEP: END
     END IF
   END IF
                                      ' もしも AH が変わったならば
                                      ' 枝番号に対するアドレスをまとめて書き込む.
```

```
    IF AH <> AH.old THEN
      J = 31
      DO
        IF branch.table(J) THEN EXIT DO
        J = J - 1
      LOOP WHILE J >= 0
      IF J >= 0 THEN
        Adr = fnADRptr(AH.old)
        V = DataArea: GOSUB PokeW
        POKE DataArea, J                 ' 枝番号の最大値
        DataArea = DataArea + 1
                                         ' サブファンクションに対する
        FOR I = 0 TO J                   ' コメントのアドレスを書き込む.
          Adr = DataArea
          IF branch.table(I) THEN
            V = branch.table(I)
            branch.table(I) = 0
          ELSE
            V = Null.Ptr
          END IF
          GOSUB PokeW
          DataArea = DataArea + 2
        NEXT
      END IF
    END IF
    READ Comment$
    V = DataArea
    IF branch THEN
      branch.table(AL) = V
    ELSE
      Adr = fnADRptr(AH)
      GOSUB PokeW
    END IF
    L = LEN(Comment$)
    IF L > 24 THEN L = 24            ' 24 バイトで打ち切る。
    FOR I = 1 TO L
      POKE DataArea, ASC(MID$(Comment$, I, 1))
      DataArea = DataArea + 1
    NEXT
    POKE DataArea, 0
    DataArea = DataArea + 1
    AH.old = AH
  LOOP
  RETURN

Int21.Comment.Code:
'        変 更 前
'                    1AE5  add   AX,86D6
'                    1AE8  cmp   AX,8786
'                    1AEB  jae   1B06
'                    1AED  mov   SI,AX
'                    1AEF  mov   SI,[SI]
'        変 更 後
DATA                 1AE5
DATA E83878        : '1AE5  call   9320
DATA 909090909090909090:                  ' 9個の nop
DATA $
DATA                 9320
DATA BED686        : '9320  mov   SI,86D6      ; もともとのテーブルの位置
DATA 3CB0          : '9323  cmp   AL,B0
DATA 7205          : '9325  jb    932C         ; ファンクション 57h 以下
DATA 2CB0          : '9327  sub   AL,B0
DATA BE5093        : '9329  mov   SI,9350      ; 拡張テーブルの位置
DATA 03F0          : '932C  add   SI,AX
DATA 8B34          : '932E  mov   SI,[SI]      ; コメントのアドレス
DATA 8A04          : '9330  mov   AL,[SI]
DATA 0AC0          : '9332  or    AL,AL
DATA 7413          : '9334  jz    9349
DATA 3C20          : '9336  cmp   AL,20        ; コメントあり
DATA 730F          : '9338  jae   9349
DATA 8A1E9A7C      : '933A  mov   BL,[7C9A]    ; AL レジスタの値
DATA 3AD8          : '933E  cmp   BL,AL
DATA 7708          : '9340  ja    934A
DATA B700          : '9342  mov   BH,00
DATA D1E3          : '9344  sal   BX,1
DATA 8B7001        : '9346  mov   SI,[BX+SI+01]  コメントのアドレス
DATA C3            : '9349  ret
DATA BEA093        : '934A  mov   SI,93A0      ; Null.Ptr
DATA C3            : '934D  ret
DATA $
'--------------------------------------------------------------
' ここに, ファンクションの番号とコメントを並べて書いておけばよい。
' 番号は 00-7Fh の範囲内とする。
' 57h 以下で SYMDEB.EXE の中で既に定義されているものも指定できる。
' サブファンクションも指定できる。
' コメントの長さは 24 バイトまでに限られる。超えると打ち切られる。
' 以下のコメント例を参考にして, 適宜自分で書き換えて下さい。
Comment.Data:
 DATA    18, "Unused"
 DATA    1B, "Get Current Drive Data"
 DATA    1C, "Get Drive Data DL(1=A:)"
 DATA    1D, "Unused"
 DATA    1E, "Unused"
 DATA    1F, "Get DS:BX Current DPB"
 DATA    20, "Unused"
 DATA    26, "Copy New PSP to segm DX"
 DATA    32, "Get DS:BX DL(1=A:) DPB"
 DATA    34, "Get ES:BX InDOS Ptr"
 DATA  3700, "Get Switch Char DL"
 DATA  3701, "Set Switch Char DL"
 DATA  4B00, "Load & Execute Program"
 DATA  4B03, "Load Overlay Program"
 DATA    50, "Set Active PSP from BX"
 DATA    51, "Get Active PSP in BX"
 DATA    52, "Get ES:BX DOS Data Table"
 DATA    53, "BPB DS:SI -> DPB ES:BP"
 DATA    55, "Create New PSP at DX"
 DATA  5700, "Get Date & Time of File"
 DATA  5701, "Set Date & Time of File"
             '                           ¦ これより短く
 DATA  5800, "Get Alloc. Strategy"
 DATA  5801, "Set Alloc. Strategy"
 DATA    59, "Get Extended Error in AX"
 DATA    5A, "Create Temporary File"
 DATA    5B, "Create New File"
 DATA  5C00, "Lock File BX"
 DATA  5C01, "Unlock File BX"
 DATA    5D, "Miscellaneous"
 DATA    5E, "MachineName,Prnter Setup"
 DATA    5F, "Redirct Mode,Assign List"
 DATA    60, "Resolve Path DS:SI>ES:DI"
 DATA    61, "Unused"
 DATA    62, "Get Active PSP in BX"
 DATA    63, "Get 2-Byte Char Table"
 DATA    64, "Get/Set Unknown Flag"
 DATA    65, "Get/Set Ext Country Info"
 DATA    66, "Get/Set Global Code Page"
 DATA    67, "Set Maximum Handle No"
 DATA    68, "Commit File(flush buffr)"
 DATA    69, "Get/Set Disk Serial No"
 DATA    6A, "Unknown Function"
 DATA    6B, "Unknown Function"
 DATA    6C, "Ext'd Create & Open File"
 DATA  FF:              '<< 終了マーク: この行を消さないこと。
'------------------------ FILE END ------------------------------
```

Windows SOURCER の世界



●著者 内田　暁 *Satoru UCHIDA(junk.35)*

The World of Windows SOURCER

はじめに

日本語Windows 3.1が出荷され，日本でも本格的なWindows時代が幕をあけたのを受け，従来から逆アセンブルツール「SOURCER」の日本語版の発売を行っていたLIFEBOATが，Windows SOURCERの日本語移植版の発売を開始した．今回移植されたバージョンは，商品名としては『Windows SOURCER V5.0』であり，動作時に表示されるマイナーバージョンはVer.5.08となっている．今回は，これについて紹介する．

The World of Windows SOURCER

Windows SOURCERの概要

『Windows SOURCER V5.0』は，従来のDOS版SOURCERに，『Windows Source』と称するWindows対応のプリプロセッサを加えた解析システム全体の呼称となっている．Windows SOURCERの製品は，ディスク2枚で供給されている．その内容を**表1**に示す．このうち，1枚目のディスクは，従来から発売されていたDOS版SOURCERのバージョンアップ部分であり，2枚目がWindows Sourceの部分となっている．

Windows Sourceが扱うものは，Windows形式の実行ファイルであるNE(Segmented-Executable)形式のファイル，仮想デバイスドライバVxDで使われる32ビットコードで書かれたLE(Linear Executable)形式のプログラム，およびWIN386.EXEに代表されるVxDの集合体として構成されたW3形式のファイルである．

これらの形式のプログラムは，DOS形式の実行ファイルに比べて，はるかに多くのシンボル情報を内包している．Windows Sourceは，同梱される各種ユーティリティを用いることによって，これらのプログラムに対してプリプロセッシング処理を行い，各種ラベル・関数名などを抽出し，定義ファイルとして書きだす．SOURCER本体は，この情報を元に，プログラムのコード解析を行うことになる．

DOS版SOURCER単体では，Windows Sourceによるプリプロセッシング処理を行った場合にのみ，Windows形式(NE形式)のプログラムの解析を行えるようになっている．そうでない場合には，「MESSAGE 49: Use Windows Source for NE portion of file.」というメッセージによって，解析を拒否されることになる．このため，SOURCERを用いてWindowsプログラムの解析を行いたい場合には，Windows SORUCERを購入することが必

写真1◆Windows SOURCERパッケージ内容



表1◆製品版ディスクの内容

●ディスク 1/2 Sourcer 5.08

README	EXE	16447	インストールユーティリティ
README	INJ	27388	インストールデータ(日本語)
README	INA	10602	インストールデータ(英語)
NEWINFO	SR5	9693	補足説明
NECDEPS	<DIR>		NEC PC-9801版
SR	EXE	199281	NEC版SOURCER本体
SDEFAULT	DEF	3410	DOSワーク情報付デフォルトDEF
NECFLOAT	TBL	7252	NEC用追加割り込みベクタ情報
TESTYN	DEF	1747	+
TESTYN	REM	5280	+ SOURCER練習用ファイル
TESTYN	EXE	843	+
INTVIEW	EXE	14299	割り込みベクタ表示ツール
JPNDEPS	<DIR>		汎用版
SR	EXE	212815	汎用版SOURCER本体
SDEFAULT	DEF	10632	DOSワーク情報付デフォルトDEF
TESTYN	DEF	1695	+
TESTYN	REM	5280	+ SOURCER練習用ファイル
TESTYN	EXE	846	+
INTVIEW	EXE	14299	割り込みベクタ表示ツール
GENERIC	<DIR>		汎用ツール
PATCH	EXE	5685	パッチ当てユーティリティ
LST2ASM	COM	17823	ファイルフォーマット変換ツール
SAMPLE1	DEF	18177	DEF形式標準(説明付き)
SAMPLE2	DEF	1154	DEF形式標準(説明なし)
HARDWARE		6640	追加割り込み情報ファイル形式
DPMI	TBL	5863	DPMI情報追加ファイル
SRMACROS	INC	4756	標準INCLUDEファイル(.ASM用)
UNPACK	EXE	10247	圧縮ファイル復元ツール
COM2EXE	EXE	5019	COM→EXE変換ツール
DOSDEPS	<DIR>		DOS依存部
DOSGEN	TBL	461	機種依存割り込み削除用ファイル

●ディスク 2/2 Windows Source 5.08

README	EXE	16447	インストールユーティリティ
README	INJ	6443	インストールデータ(日本語)
README	INA	5805	インストールデータ(英語)
NEWINFO	WSR	1397	補足説明
WINDEPS	<DIR>		Windows Source本体
EXPORTS	BAT	726	関数名一覧抽出用バッチファイル
286DOSX	DAT	5434	Phar Lap DOS Extender用データ
DOS6DLL	DAT	12728	MS-DOS 6.0 DLL用データ
OS2	DAT	32517	OS/2 1.x functions用データ
WFW	DAT	76065	Windows for Workgroups用データ
WIN31	DAT	57953	Windows 3.1 functions用データ
WINFUNC	DAT	43821	Windows 3.0 functions用データ
EXEDUMP	EXE	26950	NEファイル情報出力ツール
EXEMAP	EXE	97796	呼び出し関数一覧ツール
EXTRACT	EXE	9248	ファイルの1部分の切り出しツール
IMPDUMP	EXE	10916	ライブラリからのAPI抽出ツール
LEDUMP	EXE	24374	VxDファイル情報出力ツール
RESDUMP	EXE	28388	リソース情報抽出ツール
SYMDUMP	EXE	10592	SYSファイル中のデバッグ情報出力ツール
W3MAP	EXE	25364	W3型ファイル情報出力ツール
WINP	EXE	114180	プリプロセッサ本体
WINSMALL	EXE	12365	Windowsプログラムサンプル
SR	PIF	547	Pif-file(英語Windows3.1用)
XLAT	EXE	81412	API変換マクロ整形ツール

須となっている．仮想デバイスドライバについては，SOURCER単体でコード解析は可能であるが，この場合は，データ名などの完全な復元は行なわれない．

SOURCERでの，Windows形式の判別ロジックは，単純な“NE”，“LE”，“W3”の文字列検索であるが，これらの文字列が発見されない場合には，自動的にDOS部分の解析しか行われないので，たとえ，ターゲットプログラム中のこれらの文字列を書き換えたとしても，DOS版単体でWindowsプログラムの解析を実行することはできない．

The World of Windows SOURCER

Windows Sourceの動作．

次に，Windows Sourceの具体的な動作を紹介しよう．

Windows Sourceの本体は，WINP.EXEというプログラムである．このプログラムの基本的な動作はユーティリティプログラムローダであり，以下のような手順でツールを呼び出していく．

まず，WINP.EXEは，ターゲットプログラムをスキャンし，“NE”，“LE”，“W3”の文字列検索によってターゲットプログラムのプログラム種別を判別する．そして，そのプログラムタイプに応じて，適当な情報出力ツールを呼び出す．例えば，ターゲットプログラムがNE形式であれば，WINP.EXEは内部的にEXEDUMP.EXEを以下のような設定で呼び出している．

```
A:¥>exedump -nobanner [target] > __winp.tmp
```

(nobannerは，著作権表示を出力しないオプション)

この段階で生成されるファイルの例を**リスト1**に示す．このツールによって，プログラム形式，使用されている関数一覧などが抽出される．この段階では，関数名は，具体的な関数名には置き換えられてはいない．この他，ターゲットとなるプログラムの種別がLE形式であればLEDUMP.EXEが，W3形式であればW3MAP.EXEが，それぞれ呼び出される．

WINPは，このようにして作成された情報とWIN31.DATという名称の関数名データファイル(Windows Ver.3.1形式でないプログラムの場合にはWINFUNC.DAT)の情報(オプション設定により，データファイル名は変更可能)を元に，関数を，具体的な名称に置き換え，WDFという拡張子を持ったSOURCERへの定義ファイルのデータを作成する．関数データファイルの内容の一部を**リスト2**に，出力されるファイルの例を**リスト3**に示す．

そして，付加情報として，RESDUMP.EXEを起動し，**リスト4**に示すような形式のリソース情報ファイルを作

リスト1◆EXEDUMP.EXEの出力例

```
name ¥windows¥taskman.exe
program
target WINDOWS 3.10
80286 instructions
dgroup 2
start_csip 1:0509
start_newexe 400
description "Windows Task Manager 3.1"
modname TASKMAN

begin entry
        1       007b    1       TASKMANDLGPROC

begin segtab
        1       520     0569    CODE    USE16
        2       bc0     0060    DATA    USE16

begin modref
        KERNEL
        USER
        WINNLS

begin reloc
        1       0509    [e] start
        1       0490    [s] 1 0000      SEG
        1       0540    [m] USER 5      PTR
        1       01a5    [m] USER 262    PTR
        1       01c1    [m] USER 262    PTR
        1       0221    [m] USER 262    PTR
        1       0058    [m] USER 135    PTR
        1       031c    [m] USER 135    PTR
        1       036d    [m] USER 135    PTR
        1       0351    [m] KERNEL 158  PTR
        1       0537    [m] KERNEL 30   PTR
        1       02b1    [m] WINNLS 16   PTR
        1       02ca    [m] WINNLS 16   PTR
        1       03c3    [m] USER 286    PTR
                                .
                                .
                        (以下略)
```

リスト2◆WIN31.DATの記述例

このように，関数番号と関数名の対応表が作成されている．
公開機能については，大文字と小文字の入り混じった形で，未公開機能については，大文字のみで記述されている．

```
; ¥windows¥system¥KRNL386.EXE
; "Microsoft Windows Kernel Interface Version 3.00"
;
KERNEL . 1      FatalExit
KERNEL . 2      EXITKERNEL
KERNEL . 3      GetVersion
KERNEL . 4      LocalInit
KERNEL . 5      LocalAlloc
KERNEL . 6      LocalReAlloc
KERNEL . 7      LocalFree
KERNEL . 8      LocalLock
KERNEL . 9      LocalUnlock
```

成する．

SOURCERのプリプロセッサとしてのWINPの動作はこれで完了する．動作自体は単純なものであり，ファイル出力に要する時間も逆アセンブルに要する時間に比べれば比較にならないぐらい短く，486環境であれば1分以上かかることはまず考えられない．あとは，WINPによって作成されたWDFの拡張子をもった定義ファイルをSOURCERに読み込ませることにより，DOSプログラムと同様な解析が行われ，逆アセンブル出力結果にプリプロセッシング処理によって得られたラベル情報が自動

リスト3◆WINPの出力する.WDFファイルの例

```
uP              = P386
File format     = lst
Remarks         = except others
Code style      = New EXE
Header          = ¥windows¥taskman.exe
Keep segments   = DEF
Output filename = taskman.lst
Passes          = 5
Xref            = ON
Segment display = name
Narrow option   = ON
Vertical lines  = 0
Label type      = zero fill
Header          = Windows Task Manager 3.1
Zap load start  = 0, 0000:0c20
Input filename  = ¥windows¥taskman.exe
Math            = AUTO

 begin          end      default        seg   seg    seg
 seg:off        off     ds      es      type  size   value
 --------       ----    ----    ----    ----- -----  -----
z0052:0000      0569    z00bc   z00bc   AUTO  USE16   #1
z00bc:0000      0060    z00bc   z00bc   DATA  USE16   #2

 seg:off        type & options   label            comments
 --------       --------------   --------------   ---------
seg_2:0002        dd             dwOldSSSP
seg_2:0006        dw             pLocalHeap
seg_2:0008        dw             pAtomTable
seg_2:000a        dw             pStackTop
seg_2:000c        dw             pStackMin
seg_2:000e        dw             pStackBottom
seg_1:007b        sub,far        TASKMANDLGPROC
seg_1:0509        sub,wfarstart
seg_1:00d3        wd             FFFF:1
seg_1:00e6        wd             FFFF:2
seg_1:00f9        wd             FFFF:3

                (中略)

FFFF:33           sub,far        WAITEVENT
FFFF:23           sub,far        GETPROCADDRESS
FFFF:31           sub,far        MAKEPROCINSTANCE
FFFF:18           sub,far        FREEPROCINSTANCE
FFFF:24           sub,far        INITTASK
FFFF:32           sub,far, i 21        DOS3CALL
FFFF:29           sub,far        ISWINOLDAPTASK
FFFF:7            sub,far        ISDBCSLEADBYTE
FFFF:34           sub,far        INITAPP
FFFF:2            sub,far        GETWINDOWRECT
FFFF:6            sub,far        GETWINDOWTEXT
FFFF:13           sub,far        SHOWWINDOW
FFFF:10           sub,far        ISWINDOW
FFFF:5            sub,far        ISWINDOWVISIBLE
FFFF:20           sub,far        MOVEWINDOW
FFFF:17           sub,far        DIALOGBOX
FFFF:16           sub,far        ENDDIALOG

                (以下略)
```

リスト4◆RESDUMPによって出力されるファイルの例
この例は，-DBCSオプションを付けたもの

```
¥WINDOWS¥TASKMAN.EXE
DIALOG 10 (0Ah), "アプリケーションの切り替え"
    LISTBOX          100 (64h), ""
    DEFPUSHBUTTON    1 (01h), "切り替え(¥036S¥037ｷ)"
    PUSHBUTTON       101 (65h), "ｱﾌﾟﾘｹｰｼｮﾝの終了(¥036E¥037ｼ)"
    PUSHBUTTON       2 (02h), "ｷｬﾝｾﾙ"
    STATIC           99 (63h), ""
    PUSHBUTTON       102 (66h), "重ねて表示(¥036C¥037ｶ)"
    PUSHBUTTON       103 (67h), "並べて表示(¥036T¥037ﾅ)"
    PUSHBUTTON       104 (68h), "ｱｲｺﾝの整列(¥036A¥037ｾ)"

VERSIONINFO 1 (01h)
    VS_VERSION_INFO
    File Version 3.10.99
    Product Version 3.10.99
    Operating System: Windows 16-bit
        Type: Application
```

リスト5◆EXEMAPによる出力例

```
Copyright (c) 1992-93 Andrew Schulman.  All rights
reserved.

/* 1:0000 */ start() {
/* 1:0001 */     INITTASK();
/* 1:0025 */     LockSegment();
/* 1:003c */     WAITEVENT();
/* 1:0045 */     INITAPP();
/* 1:0057 */     GetWinFlags();
/* 1:00b2 */     UnlockSegment();
/* 1:018c */     TranslateMessage();
/* 1:0196 */     DispatchMessage();
/* 1:01a6 */     GetMessage();
/* 1:01e5 */     LoadIcon();
/* 1:01f4 */     LoadCursor();
/* 1:01fe */     GetStockObject();
/* 1:021d */     RegisterClass();
/* 1:0258 */     CreateWindow();
/* 1:026a */     ShowWindow();
}
```

的に付加されることになる．

ここで，Windows Sourceでは，多種のVxDの集合体であるW3形式のプログラムについては，全体を一度に解析することはなく，その中の一つのモジュールを選択し，解析するという仕様になっている．例えば，W3形式の代表格のWIN386.EXEは非常に巨大(約600Kバイト)なファイルであり，ファイル全体の逆解析を一度に行うことは現実的でない(というより，メモリなどの関係から不可能である)ため，このような仕様になっていると思われる．余談になるが，NEC MS-DOS 5.0A添付のDPMI.EXEをWINPにかけると，WIN386.EXEとまったく同様の表示となる．このようなツールを用いれば，DPMI.EXEとWIN386.EXEが事実上同一のものであることは，一目で確認できるようになる．

The World of Windows SOURCER

その他のツールについて

Windows Sourceには，プリプロセッサ本体のWINP以外にも，いくつかのツールが付属している．

EXEMAP.EXEは，**リスト5**のような形式で，プログラム中に含まれるAPI呼び出しを，未公開のものも含めて，セグメント・オフセット順に出力するものである．これは，逆アセンブル処理，すなわち解析処理を行ったものではなく，あくまでも，ただ単純にセグメント・オフセット順に出力されただけのもので，呼び出しの親子関係については，なんら保証されていない．しかしながら，呼び出される関数の一覧が手っ取り早く作成でき，また出力されるファイルサイズも比較的小さいため，プログラムの雰囲気を大ざっぱにつかむのには，非常に便利なツールである．

EXPORTS.BATは，指定されたファイル群中に含まれる関数一覧をデータファイルとして出力するためのツールである．通常は，指定ディレクトリを¥WINDOWS¥SYSTEM¥＊.EXEとして実行することで，データファイルを最新の状態にするために用いることになる．

XLAT.EXEは，V86MMGRが提供するAPI変換マクロに対応して，フラグメントデータ(dbで記述されたリスト出力)を，逆アセンブルリストに対応させるものである．SOURCERは，デフォルトではこのAPI変換マクロへの記述に対応していないため，解析を行う場合には，手動で変換マクロ記述部分のアドレスを見つけだし，その部分についてこのツールを用いて，リスト変換を行うことになる．

The World of Windows SOURCER

Windows Sourceの性能

Windows Sourceで行っている内容は，あくまでもWindowsプログラム中のシンボル情報の抽出のみであり，複雑なロジックによる解析を行っている訳ではない．したがって，このツールの動作は，他の同機能ツールと比べて，原理的な差別化を図ることは不可能である．実際，逆アセンブルツール『Agnes』(美利崎人)では，Windows Sourceと同様な内容を持った関数データ

ファイルを持つことにより，特に専用のプリプロセッサを用いることなく，DOSプログラムと同様の操作で，Windows SOURCERと同等の解析結果を得ることができる．また，『DNA98』（マイクロデータ）でも，データファイルを持たないため，外部関数名の復元は行わないが，解析対象プログラム自身の持っている関数名のASCII名復元は実現されている．

その意味から，Windows SOURCERの本質的な優位性は，解析者にとって使い慣れたシステムであるSOURCERに，Windowsアプリケーションの情報を渡すことができる，という1点にとどまることになる．

しかしながら，その中にも，いかにもSOURCERらしいユーザインターフェース上の差別化は行われている．具体例としては，ユーティリティが統合ツールとしてまとめられておらず，分割されたプログラムとして供給されていることにより，手慣れた解析者にとっては，それぞれのツールを独立に用いることにより，より効率的な解析が行えること，また，ディスク上のプログラムから関数名を抜き出し，データファイル化するツールが付いていることにより，関数名データファイルを，常に最新の状態に容易にアップデートする手段が提供されていること，添付の関数データファイルについて，公開関数はGetVersionのように，大文字/小文字の混ざった形式で，未公開関数はEXITKERNELのように，大文字のみで記述され，容易に識別可能になっていることなどが挙げられよう．

The World of Windows SOURCER

Windowsアプリケーションの解析の実態

Windowsプログラムの世界では，もはやアセンブラレベルでの「完全なソースの復元」は，ほとんど無意味だと言っていい．仮想デバイスドライバ部分を除くアプリケーションのほとんどすべてが，Cなどの高級言語で書かれ，各種ライブラリを用いることが前提となっている以上，アセンブラレベルに落としたところで，そのプログラムの保守維持は不可能であるし，そもそもソースリストが膨大になりすぎ，ほとんど処理不可能になる．

したがって，Windowsプログラムの世界での逆アセンブルツールの役割というものは，プログラムの構造の分析・使用されている関数の抽出，およびコアとなる部分の動作解析にほぼ絞られるといっていいだろう．

また，Windowsプログラムの解析の精度をあげるためには，SDKなどに収録されている，システムの各部分についての可能な限りのソースコードを入手することが不可欠である．とくに，各プログラムのヘッダファイルに記述されている，データ構造情報などは，解析を行う意味からも，極めて重要である．Windowsプログラム解析は，膨大なデータを，いかに適切に収集し，効率良く調べていくかということに尽きるといってもいいだろう．

DOSプログラムとWindowsプログラムを比べたときに，大きな差となって現われてくるのは，解析に必要な情報量の圧倒的な差である．Windowsプログラムは，内部で多くのWindows APIを呼び出しており，解析に必要なクロスリファレンスは，DOSプログラムとは比較にならないほど多い．その意味からも，解析環境のマルチウィンドウ化は，必須ともいえる．

しかしながら，Windows上での解析環境は，現時点ではとても良好であるとは言えない．Windows SOURCERのマニュアルでは，SOURCERのWindows DOS互換BOXでの動作についても触れられているが，解析対象アドレスをリアルタイム表示するという仕様のため，解析中に非常に多くの情報量を画面表示するSOURCERは，画面表示の遅いDOS互換BOXでは，マルチタスク動作に起因するDOS互換BOX中でのCPU割り当ての低減分以上に，極端に動作速度が低下する．

Windows世代となり，膨大な情報量のものを解析する必要が出てきた現在，逆解析ツールも，Windows上で動作するものに変貌していく必要性が強く感じられる．また，DOSに縛られているための，シンボルテーブルサイズなどの各種ワーク領域サイズにも制限が出てきており，Windows時代の解析ツールとしては重い足枷となっていることも，解析ツール自身のWindowsプログラム化を促す理由となるだろう．

The World of Windows SOURCER

SOURCER本体の前バージョンとの違い

今回のバージョンアップで，SOURCER本体（コード解析部分）の機能向上も行われている．主な強化項目を**表2**に示す．今回のバージョンアップでは，前回の

Ver.3.18からVer.4.52へのバージョンアップにおける，PC-98系マシンでの実行速度の5倍以上の向上のような，大きな目玉となるような項目はない．確かに，実行速度も10～30%程度速くなっているし，例えばCOM形式プログラムの最初にjmpコードがあった場合，その飛び先にreal_startというラベルが付けられるようになったなどのように，出力ファイルへのコメント項目・サブルーチンへのラベルの付け方などがより洗練されてきたなどの改善点は確かにある．しかし，総合的な解析環境としては前バージョンと感覚的に差は感じられない．また，解析能力・および解析オプションにもほとんど変化はない．日本版SOURCERの今回のバージョンアップの目玉は，あくまでも，Windows Source部分がLIFEBOATから初めて提供されたということに尽きるだろう．

また，今回のバージョンアップで，**表3**に示すようなツールが新たに添付されている．これらは，確かに逆アセンブル時の補助ツールとして用いると便利なものではあり，このようなものが添付されることは素直に評価すべきものでもあるが，いずれも，ごく基本的なツールであり，プログラム解析を本気で行うレベルの人であれば，この程度のツールは既になんらかの手段で入手していると思われ，実用上の意味はあまりないだろう．特に，パッチツールについては，ファイルサイズから想像されるとおり，コマンドライン入力型の原始的なものであり，全画面対話型のツールが常識となっている最近のアプリケーションの風潮からすると，あまりに使いにくく，実用ツールとしての効果は疑問である．

現在，あらゆるあらゆるジャンルで，DOSで動作するアプリケーションには，行き詰まりが見られる．逆アセンブルツールというジャンルでも，他のアプリケーションと同様，MS-DOSという狭い舞台で行えることは既にやり尽くしてしまったといえるだろう．

The World of Windows SOURCER

日本語化について

今回のSOURCERでは，新たに日本語化されたWindows Source部分については，十分な日本語化が行われているとはいえない．

まず，第1に挙げられるのはWINP.EXEでのデータ自動抽出では，リソースファイルの生成を2バイト文字にする方法が存在しないことである．リソースデータ抽出プログラムRESDUMP.EXE自身は，-dbcsオプションという，2バイト文字対応化オプションを持っているため，日本語化されたリソース情報が欲しい場合には，マニュアルで実行すればいいのではあるが，日本語版と銘打つ以上，WINP.EXE自身にも-dbcsオプションを追加し，日本語対応を行って欲しかったところである．

第2に，提供されたディスクに含まれる関数データファイルは，すべて米国版のWindowsなどから生成されたものとなっている．したがって，日本語Windows用に開発されたアプリケーションの解析では，日本版Windowsから直接APIを生成することになる．これによって，せっかく製品版のデータファイルで行われている，公開API，未公開APIの識別方法が，ほとんど無意味となってしまう．次回バージョンでは，ぜひ日本語Windowsに対応したデータファイルの添付を行ってもらいたい．

3点めとしては，デフォルトで提供されているSR.PIFが，英語版Windows用のみとなっていることが挙げられる．日本版と称する以上，日本語Windows用のPIFも標準で添付すべきだろう．

また，今回SOURCERでは，DOS/V日本語モードでの画面直接描画をサポートしたとしているが，DOS標準添付の$DISP.SYS，およびこれのマイナーチェンジである$DISPH.SYSのビデオモード3では確かに正常に表示が行なわれたが，DISPV.SYS，DISPS3.SYSなどのツ

表2◆SOURCER本体のバージョンアップ項目

- 解析速度の向上(10～30%)
- DOS/V日本語モードでの画面直接描画のサポート
- DOS/V BIOSのサポート
- Pentium Processerコードのサポート
- 対応アセンブラの強化

表3◆SOURCER V5.0での新規サポートツール

UNPACK	MicrosoftのEXEPACKで圧縮されたプログラムの復元を行う．
COM2EXE	マルチセグメントCOMファイル(COMファイル形式で，プログラム開始時にEXEファイルのローダ部分を呼び出すことにより，実質的にEXEファイルの構造を持ちながら，形式上はCOMファイルとなっているもの)を，通常のEXEファイルに復元する．
INTVIEW	割りこみベクタ内容一覧ツール
PATCH	バイナリパッチツール

ールのビデオモード3では画面表示が行われなかった．画面表示ロジックの問題と思われるが，これも解決すべき問題であるといえるだろう．

The World of Windows SOURCER

マニュアルについて

Windows SOURCERのマニュアルは1冊にまとめられ，前半がDOS版SOURCERの機能説明，後半がWindows Sourceの機能説明となっている．

前回のバージョンの時点で，SOURCERのマニュアルは，（やや専門的すぎるきらいはあったものの）丁寧でできのいいものであったが，今回新たに日本語版の起こされたWindows Sourceの解説部分は，DOS版部分にひけをとらないほど丁寧かつ詳細なものとなっている．

このマニュアルでは，まるまる30ページにわたって「Windowsプログラムの読み方」というタイトルで，実際の解析手順に沿った形でWindowsプログラムの逆アセンブリリストの解読法について説明が行われている．さらに，仮想デバイスドライバVxDについても，25ページにわたって，その構造・解析手法について書かれている．前者は，実質上のプログラム起点である“WinMain”の発見，ウィンドウ手続き・ダイアログ手続きに始まる解析の手法，リソースファイルとの関係，関数呼び出しの読み取り方などについて，製品添付のサンプルプログラムWINSMALL.EXE，およびMS-Windows製品中の最小のプログラムTASKMAN.EXEなどを例にとって詳細かつ徹底的に解説したものであり，後者はVxDの構造，VxDが含まれる可能性のあるファイルの例，解読の要点となるプログラムコードの解説などについて記述している．

ここに書かれている内容は，自力でWindowsプログラムを書いている人にとっては，ほとんど自明のことともいえるレベルのものがほとんどであるが，Windowsプログラム解析は，DOSの世界とはまったく異なった発想のものとなる以上，この程度の説明はやはり必須のものだろう．

考えようによっては，Windowsプログラムの解析ツールの提供という点よりも，Windowsプログラム解析の入門書ともいえる，このマニュアルの品質こそが，Windows SOURCERの売りだといえるかもしれない．

The World of Windows SOURCER

おわりに

SOURCERのDOS部分（コード解析部分）については，今回も目立った改善点もなく，現状でほとんど完成したものになっている．また，今回初めて日本語化されたWindows Sourceの部分は，動作としては単なるプリプロセッシングを行っているだけであり，日本語化が不十分である点を除けば，本質的なバージョンアップ項目は特に考えられない．その意味で，DOS上で動作する解析ツールとしてのSOURCERという製品は，そろそろ最終的な形態となったといえる．

そして，DOS版としての完成は，同時にDOS上での動作の限界をも示唆している．生DOS上では統合環境的機能を持たず，Windows DOS互換BOX上では動作速度の落ちる現在のSOURCERでは，Windowsプログラム解析には，どうも力不足の感がある．

エンドユーザが，どれだけWindowsを実用環境として使っているかどうかは別問題として，アプリケーションの新規開発は否応なしにWindowsを中心とした環境に移行している．そのような状況のなかで，近い将来逆アセンブルツールの世界にも，Windows化の動きが出てくるのではないだろうか．

現在，リバースエンジニアリングそのものの適法性も問題となっているが，このような問題を解決し，本格的なプログラム解析環境が完成していくことを期待したい．



製品名	Windows SOURCER V5.0
対応機種	PC-9800シリーズ，各社DOS/V機（OADG），J-3100シリーズ，FMRシリーズ，PC/AT
必要メモリ	メインメモリ640Kバイト以上（最大6MバイトのEMSを使用可能）
価格	5万8千円
連絡先	㈱LIFEBOAT TEL 03-3293-4711　FAX 03-3293-4710

本格派翻訳ソフトの出現!!

最高峰をいく LogoVista E to J for Windows

著者●大場五夫 *Itsuo Ohba*

◪プロローグ

パソコン用英日翻訳システムもすでに数種類が発売され，狭いシェアの中で，激しい競争を繰り広げています．そうした中で，従来Macintosh用だけが発売されていた「LogoVista E to J」(以下LogoVistaと略)がPC-9801シリーズのWindows3.1版として発売され，注目を集めています．

LogoVistaは，Macintosh用が発売された当初から，現在，日本に存在する英日翻訳システムの中では，最も優れたシステムだという評判を耳にしていましたが，実際にWindows版を使ってみて，その評判が嘘でなかったことを実感しました．

LogoVistaの優れた点は何といっても，その構文解析能力の優秀さです．多くの翻訳ソフトが誰の構文解析システムを採用しているのか，明らかにしていない中で，このLogoVistaだけは，言語学者として世界的に有名なハーバード大学教授である久野 暲(くの・すすむ)氏が開発した構文解析システムを採用していることを明示しています．

大型コンピュータ用やワークステーション用の翻訳システムを含めて，現存のほとんどの翻訳システムは，何らかの前処理を施さない限り，満足すべき翻訳結果は出力しません．しかし，このLogoVistaだけは，前処理を行わなくても，他の翻訳システムで前処理を施したのと同じ程度の翻訳結果を出力する点が特徴です．

このような本格的な翻訳システムが20万円を切る価格(194,000円)で発売されたとなれば，数百万円もするような翻訳専用機は出る幕がなくなることだけは確かです．

しかし，LogoVistaはパソコン用のソフトとしては，かなり大規模なもので，総容量は35Mバイト，そのうちメイン辞書だけでも約19Mバイトというボリュームです．

そのほかにWindows 3.1が約31Mバイトを必要としますので，この両方を合わせれば，優に80Mバイトのハードディスクを占拠してしまう結果となります．さらにLogoVistaは，もともとワークステーション用に開発されたものですので，パソコンで使用する場合は本格的な486マシンでなければ，処理速度の面で難点があります．

そのような意味で，以下にLogoVistaの実力のほどを細かに見てみたいと思います．

◪起動画面と基本的アーキテクチャ

まず**図1**を見てください．これはLogoVistaの起動時の画面です．

LogoVistaはWindows版なので，他のWindows版ソフ

トがそうであるように，ほとんどの処理をマウス操作で行うように作られています．したがって，キーボードを使う場面は，ファイル名を入力するときか，字句を修正する場合，辞書登録の場合程度に限られており，その面でも親しみやすく操作性も優れています．

LogoVistaでは，翻訳を行う前に翻訳条件を決めなければなりません．その機能は**図1**の［オプション(O)］のメニューに用意されていますので，これをクリックして開いてみます．**図2**に示す翻訳オプション設定メニューが開かれます．

この画面で先頭の○をクリックすれば，その機能が選択されて●に変わります．翻訳オプションの中で特徴的なものは，固有名詞の扱いと翻訳速度を選択できるという機能です．

固有名詞の扱いを［英語のみ］で翻訳すると，例えば英語の文中にDNAという略語が出てきた場合，翻訳結果にもDNAだけが出力されます．しかし，［日本語(英語)］を選択して翻訳すると，翻訳結果は「デオキシリボ核酸(DNA)」というように，先に日本語訳を出力したあとに，原文のDNAをカッコ付きが表示されます．そして，［日本語のみ］を選択すると，翻訳結果は「デオキシリボ核酸」だけが出力されてDNAは出力されません．

このように固有名詞の扱いをきめ細かに指定できるのはLogoVistaだけであって，他の翻訳ソフトには備わっていない機能です．

さらにもう一つの特徴である［速度選択］は0から100まで小刻みに速度を指定することができます．そして，翻訳速度と翻訳精度は表裏一体の関係で連動しています．つまり，翻訳速度を最も遅く(0)設定すれば，処理時間はかかるけれども，翻訳精度は最も優れたものとなります．それに対して翻訳速度を最も速く(100)設定すれば処理時間は速くなる代わりに翻訳精度が，ガタっと低下します．

筆者は，この翻訳速度を5刻みで同じ英文を何段階かに分けて翻訳させてみましたが，翻訳精度は中間点の49と51を境目(ジャスト50という数字は欠番)にして，0から49までがよい方，51から100までが悪い方というように大きく分類することができます．

しかし49以下の速度であっても，原文の構文次第で正しい翻訳を出力しないことがありますので，翻訳速度は25ないし30程度に設定するのが実際的な考え方のようです．事実，筆者のテストでは，0から25までの範囲では，翻訳時間も翻訳精度もほとんど差違が見られないという結果が出ています．

このようにLogoVistaは，翻訳精度が優れているだけでなく，翻訳に関するあらゆる面できめ細かい配慮が行き渡っているのが特徴といえるでしょう．特にここに上げた固有名詞の扱いと翻訳速度の選択機能は，他の翻訳ソフトの追随を許さぬ優れたアーキテクチャです．

◪ファイル読み込みと翻訳の実行

翻訳を行うには，翻訳すべき英文がなければなりませんので，まず英文を読み込んでみましょう．この機能は**図1**の［ファイル］に用意されていますので，これをクリックしてみます．**図3**に示すメニューが開かれます．

翻訳する英文を新しく入力する場合は，このメニューから［新規作成］を選択しますが，すでに英文原稿が用意されている場合は，このメニューから［英文テキス

図1●LogoVista E to Jの起動時画面

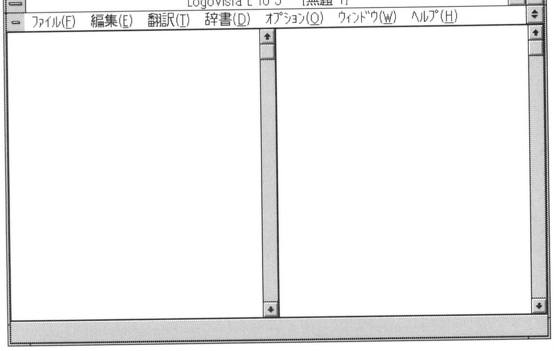
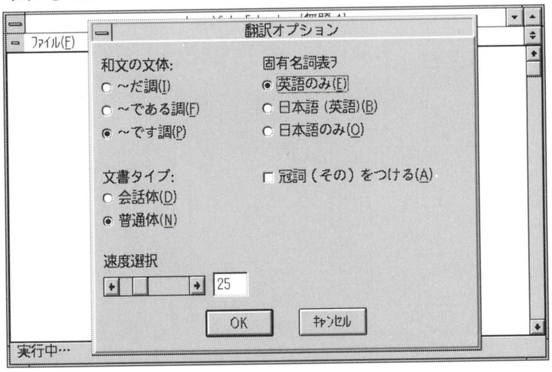


図2●翻訳オプション設定画面



トの読み込み]を選択します．これをクリックすると，**図4**に示すようにファイル名の入力欄とディレクトリを選択する画面が表示されます．

なお，このメニューの中の[テキストの書き込み]という機能は，翻訳結果をMS-DOSの標準テキストに変換するための機能です．

筆者の場合はドライブBの「EJ」というディレクトリに「TESTTXT.ENG」というテスト用の英文が用意されています．このような条件を設定して[OK]をクリックすると，即座に指定した英文が読み込まれ，**図5**のように表示されます．

ここで翻訳を実行する場合は，[翻訳]メニューを開いて[翻訳]をクリックすると，即座に翻訳が開始されます．しかし，その前に[未知語検索]を実行するのが実際的な使用方法でしょう．

この機能を使うと，原文の中にシステム辞書やユーザ辞書に含まれていない単語がある場合，それらをリストアップしてくれますので，翻訳する前にこれらの単語を辞書登録することにより，二度手間を省くことができるからです．

未登録単語の辞書登録を終わって，翻訳を実行させると，**図6**に示すように翻訳の進行状態が表示されます．

翻訳結果はセンテンス単位(ピリオドまで)で出力されますが，翻訳対象のセンテンスが変わるごとに進行状態の表示も変わります．そして全文の翻訳が終了すると，**図7**に示すように翻訳結果全体が画面に表示されます．この翻訳結果を見ると，一応正しい直訳を出力していることが分かります．

機械翻訳はどのみち直訳しかできませんので，正しい直訳が出力できるということは，それだけでもLogoVistaの実力が分かろうというものです．現実には，同じ英文を既存の他の翻訳ソフトで翻訳させても，これだけの結果は出力できません．

ただし，この出力結果のうち，eukaryotic gene(真核遺伝子)，E-coli(大腸菌)，recombine(再結合する)の三つの用語については，事前に[未知語検索]機能でリストアップさせ，辞書登録を行っておきました．また，Thereforeの翻訳結果もオリジナルでは「それゆえに」と翻訳されますが，これもあえてユーザ辞書に「したがって」という訳語を登録した結果です．

図3●ファイル処理メニュー

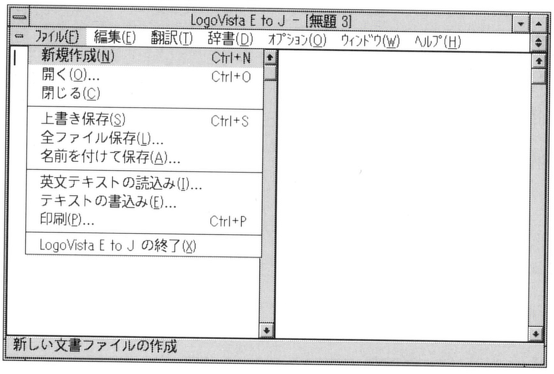


図5●読み込まれた英文と[翻訳]メニュー

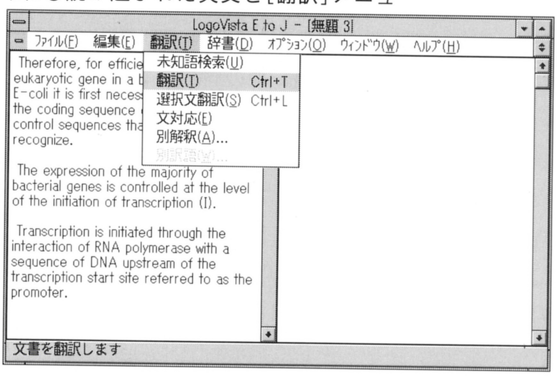


図4●ファイル名とディレクトリの指定画面

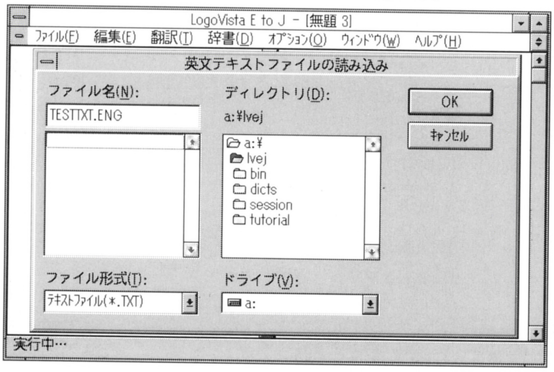
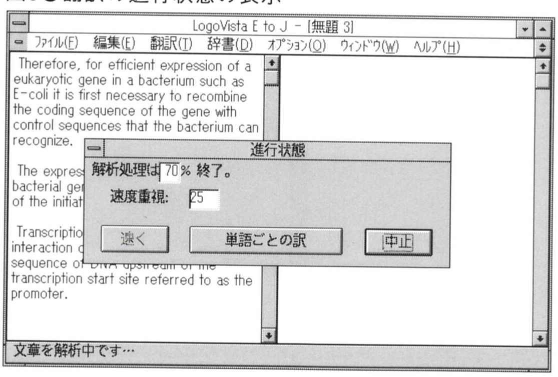


図6●翻訳の進行状態の表示



◪翻訳速度による翻訳精度の相違

図7の翻訳結果は，翻訳速度を25に指定して行った結果ですが，翻訳速度を速くすることによって，どの程度翻訳精度が低下するかの実例を見てみましょう．

図8は翻訳速度を51で翻訳したものです．これで見ても分かるように，第1センテンスは，速度を上げた結果，明らかに構文解析を誤っていることが分かります．それに対して第2センテンスと第3センテンスは，図7と同じ結果を出しており，単純な構文では，多少の翻訳速度の違いは翻訳結果に影響を与えないことを示しています．

しかし，**図9**に示すように一挙に翻訳速度を100に引き上げると，全文が正しい翻訳を出力しないようになってしまいます．**図8**と**図9**で「‖」マークはシステム自体が構文解析に失敗したことを示すマークです．

このように翻訳速度をコントロールすることによって，翻訳精度をコントロールできるようにしたアーキテクチャは，コンセプト自体もユニークですが，これは翻訳システムの生命ともいうべき構文解析機能が，いかに処理速度（CPUの性能）と密接な関係を持っているかを理解する上でも，貴重な側面です．

翻訳システムの一定時間あたりの構文解析機能は，CPUの速度に左右されるので，それを活用することによって優れた構文解析機能を発揮させられます．

しかし，その実体は多岐に枝分かれしたツリー構造や何重にも積み重ねられたループ構造ですから，翻訳精度をコントロールできるということは，それらの多重度を端折ることによって構文解析の精度そのものも端折ることができることを意味しているからです．

◪多彩な翻訳機能

LogoVistaでは，いったん最初の翻訳を終了すると，さらに多彩な翻訳機能を使うことができます．**図10**を見てください．

ここに示した［翻訳］メニューは**図5**に示したメニュ

図7●テスト文書の翻訳結果

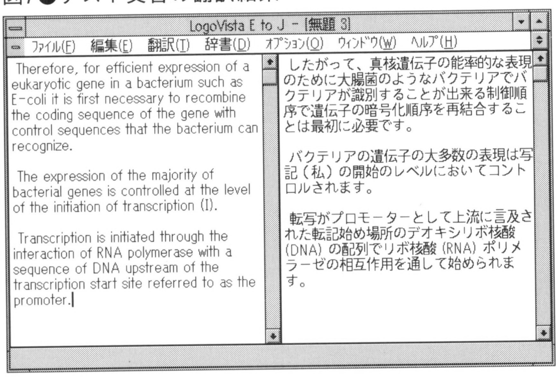


図8●同じ文書を翻訳速度51で翻訳した結果

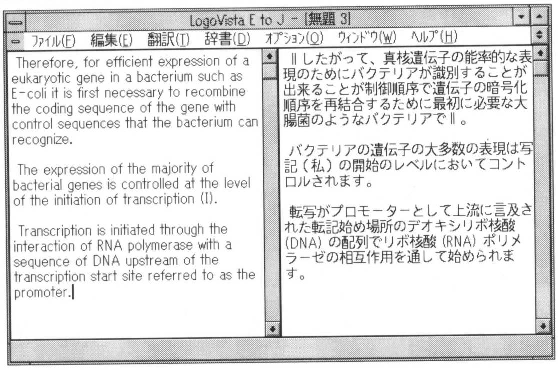


図9●翻訳速度100の翻訳結果

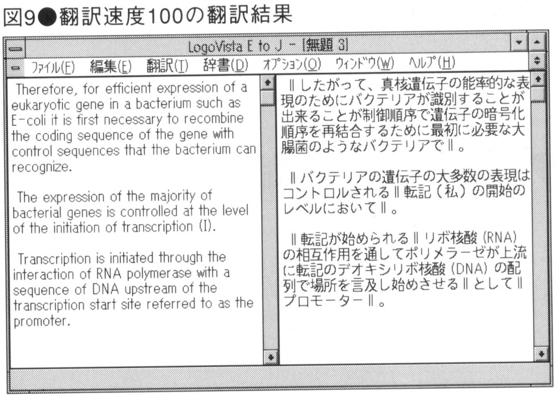
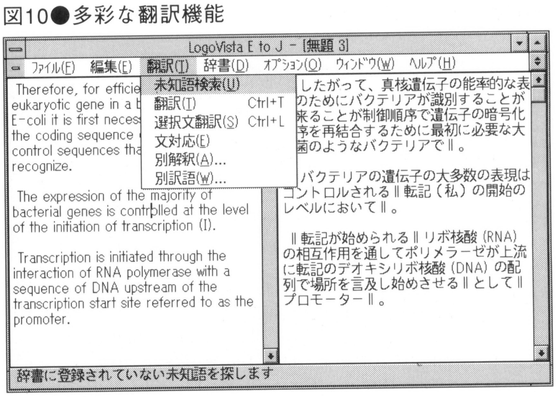


図10●多彩な翻訳機能



ーと異なっています．**図5**では翻訳機能は3項目しか用意されていませんでしたが，**図10**ではそれが倍の6項目に増えていることが分かります．

追加された機能としては[選択文翻訳][文対応][別訳語]の三つがありますが，これらはいったん翻訳を実行した後でなければ使用できない機能です．このうち[選択文翻訳]は翻訳結果を見直して，改めて辞書登録などを行ったあと，1センテンス単位で再翻訳させる場合に使用します．

例えば**図10**の第2センテンスで(私)という訳語がありますが，これはローマ数字の「1」を表すのが本来の意味ですので，原文の(I)をまとめて「(I)」と登録しなおし，改めて翻訳しなおす場合などに使用します．

それに対して[別解釈]の機能は，選択した単語単位で別の訳語と入れ替える機能です．例えば，第2センテンスの中で「is controlled」が「コントロルされる」という訳として出力されていますが，この訳語としては別に「制御される」「調節される」「統制される」などが用意されていますので，それらの中から適訳と思われる訳語を表示させて，入れ替えることができます．

いずれにせよ，こと翻訳に関する限り，ユーザの多様な要請に応えられるよう万事柔軟性を持ったアーキテクチャになっている点に好感が持てます．

辞書登録機能

翻訳システムの優劣は，第1に構文解析機能の性能，第2に辞書の充実度で判断されますが，LogoVistaは，そのどちらをとっても，他の追随を許さぬだけの内容を備えています．

構文解析機能についてはすでに述べましたので重複は避けますが，辞書登録機能は，翻訳画面から個別に登録していく方法と，システムに標準添付されたユーティリティを用いて，一括登録する方法の2種類が用意されています．

このうち一括登録の方法は，条件を細かく指定することによって，どんなケースにも対応できるように構成されている点が特徴で，この機能をフルに活用すれば，システム自体を賢くしていくことができ，それによっていっそう優れた翻訳をすることができます．

その具体例はスペースの都合上，割愛せざるを得ませんが，翻訳画面からの個別登録については具体例で説明しましょう．

辞書登録機能は[辞書]に割り当てられているので，これをクリックすると，**図11**に示す画面が開かれます．登録できる品詞は，名詞，動詞，形容詞，副詞の4種類ですが，名詞の場合は，**図11**に示すように英語と，複数形，日本語訳の3項目を入力するだけで[登録]をクリックすれば，それだけで登録作業は終了します．

動詞の場合は，より簡単で，英語欄に単語を入力して[○動詞]の○をクリックされば，過去形，三人称単数，現在分詞の3項目は対応する形が自動的に表示されますので，日本語訳を終止形で入力するだけで登録することができます．

エピローグ

以上，LogoVistaの主要な機能を紹介しましたが，筆

図11●名詞の登録画面

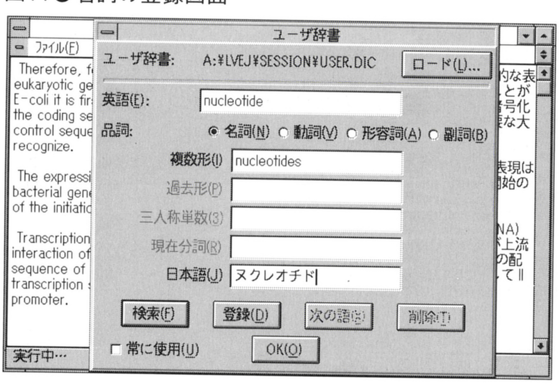
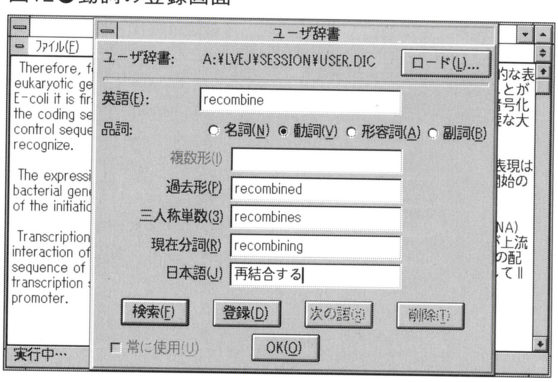


図12●動詞の登録画面



者はこれまでにも，ありとあらゆる翻訳システムを試用し，そのレポートを書いてきました．その経験に照らしていえば，機械翻訳で完璧が期しがたい以上，LogoVistaは現時点で望みうる最高水準の機能を備えたものということだけは明言できます．

今後は恐らく，すべての翻訳システムがLogoVistaの域に迫ることを目標にして改良を加えていくことになるのでしょう．その意味からも，筆者のような翻訳ユーザは，このような優れた翻訳システムが出現したことを素直に喜びたいところです．

ところで，翻訳システムは肝心の構文解析機能と辞書機能が充実していれば，あとはどうでもいいと思われがちですが，それでもやはり，編集機能や印刷機能，翻訳結果をテキストに変換する機能など，最低限備えるべき機能は備わっていなければなりません．それらの面でのLogoVistaの機能は，おおむね満足すべきレベルにはありますが，あえて不満を漏らすとすれば，編集機能が弱いということでしょう．

そのため現状では翻訳結果を修正する場合は，テキストに変換したものを他のワープロやテキストエディタに読み込んで修正するという形をとらざるを得なくなります．

編集機能については次期バージョンでぜひとも強化してほしいところですが，もう一つ改良してほしい点をつけ加えれば，英文領域に全角文字を混入させることができるようにしてほしいという点です．

それはなぜかというと，技術文書では，α，β，γ，λ，μ，Σ，Φなど，半角文字には含まれないギリシャ文字や特殊記号がふんだんに使われるからです．また30℃などの摂氏温度表示なども半角文字では表示できません．

現状では，これらの全角文字が英文領域に混入した場合，「^A」「^B」「^K」のように文字化けを起こしてしまい，その部分が翻訳精度に悪影響を及ぼす結果となります．

それ以外の機能，例えば印刷機能，テキスト変換機能（ともにファイルメニューの中），ヘルプ機能などは，ワープロ並みに充実しているので，編集機能が弱いということは何かアンバランスな，ちぐはぐな感じを否めません．

中でもLogoVistaのヘルプ機能は強力で充実しているので，ほとんど印刷物のマニュアルを必要としないほどです．これらの機能については，画面だけを表示しますので参考にしてください．

LogoVista E to J Ver.1.2（Windows3.1対応版）

●開発元：ロゴヴィスタ株式会社

●発売元：カテナ株式会社　ロゴヴィスタ事業部

問い合わせ　☎03-3615-9015

図13●印刷機能のメニュー

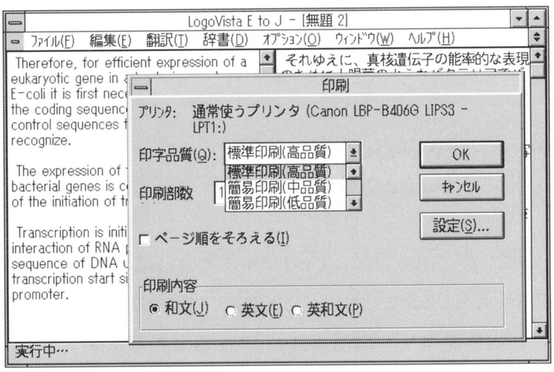


図14●ファイル書き込み(テキスト変換)機能の画面

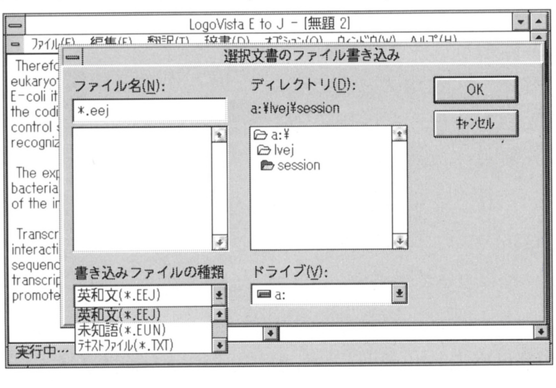
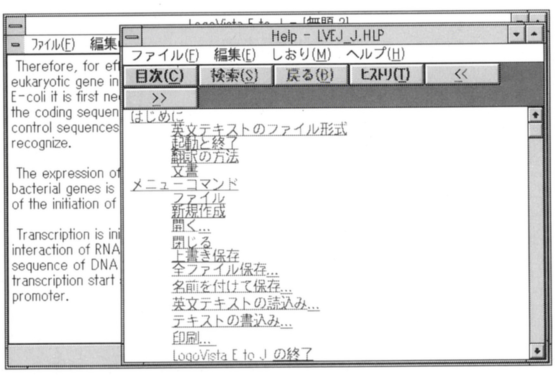


図15●オンラインヘルプの目次





i486を使ったPC-9801RA/DA用CPUアクセラレータ

SS486の製作

●著者 甲村　実　Minoru KOHMURA

i486(ODPを含む)を使用した，PC-9801RA/DA用のCPUアクセラレータ"SS486"を作りましたので，ここに紹介します．

SS486の回路構成の説明

回路図(稿末)には，CPUは487と書いてありますが，CPUは，i486SX，ODP，i486DX，i486DX2の4種類共に使用可能です．PGAタイプの各CPUのピンの比較表を**表1**に示します．また，合わせてi386DXのピンの機能も示しておきます．

なお，i386DXとi486系はPGAのパッケージの大きさが違います．386DXは14ピン角で，i486系は17ピン角，P24Tは19ピン角です．

i486用のスルーホールの周りに，一周り余分のスルーホールを空けてあります．これは，P24T用のスルーホールです．ただし，現在のところP24Tの実物が発売されていないので，動作確認はできていません．また，発熱や消費電力や処理速度などの検討項目があります．

ハード上は，i486DXはi386にコプロセッサとキャッシュを内蔵させて高速化した構造になっていますから，同名の信号内容の機能はほとんど同じです．

i386DXとi486系では，アドレスやデータの本数は同じです．コプロセッサが外付けか内蔵かで数ピンの制御信号が違っているのと，内蔵キャッシュのための制御信号が，i486では増えていることが大きな違いです．

i486系同士の同名信号内容のピンの機能は同じですが，物理的な配置の異なっているピンが数ピンあります．SS486の使用方法で問題になるピンは，NMIの部分だけです．そのため，NMIはディップスイッチで切り替えるようにしています．PC-98では，NMIはパリティエラーなど特別な用途に使用されています．そのために，通常はほとんど動作していません．コプロセッサ関係は，i386DXの場合でもコプロセッサと直接接続されていたので，それが内蔵された場合には，特別な外部での処理は必要ありません．

それと，もう一つ大きな違いはクロックにあります．i386DXは，動作周波数の倍のクロックをCPUに入れる仕様になっています．i486は，動作周波数と同じクロックを入れるようになっています．このため，単純にi386DXとi486の同じ機能のピン同士を接続しただけで

信号名	386DX	486SX	487SX	ODP	486DX	486DX2
A2	C4	Q14	Q14	Q14	Q14	Q14
A3	A3	R15	R15	R15	R15	R15
A4	B3	S16	S16	S16	S16	S16
A5	B2	Q12	Q12	Q12	Q12	Q12
A6	C3	S15	S15	S15	S15	S15
A7	C2	Q13	Q13	Q13	Q13	Q13
A8	C1	R13	R13	R13	R13	R13
A9	D3	Q11	Q11	Q11	Q11	Q11
A10	D2	S13	S13	S13	S13	S13
A11	D1	R12	R12	R12	R12	R12
A12	E3	S7	S7	S7	S7	S7
A13	E2	Q10	Q10	Q10	Q10	Q10
A14	E1	S5	S5	S5	S5	S5
A15	F1	R7	R7	R7	R7	R7
A16	G1	Q9	Q9	Q9	Q9	Q9
A17	H1	Q3	Q3	Q3	Q3	Q3
A18	H2	R5	R5	R5	R5	R5
A19	H3	Q4	Q4	Q4	Q4	Q4
A20	J1	Q8	Q8	Q8	Q8	Q8
A21	K1	Q5	Q5	Q5	Q5	Q5
A22	K2	Q7	Q7	Q7	Q7	Q7
A23	L1	S3	S3	S3	S3	S3
A24	L2	Q6	Q6	Q6	Q6	Q6
A25	K3	R2	R2	R2	R2	R2
A26	M1	S2	S2	S2	S2	S2
A27	N1	S1	S1	S1	S1	S1
A28	L3	S1	R1	R1	R1	R1
A29	M2	R2	R2	R2	R2	R2
A30	P1	P3	P3	P3	P3	P3
A31	N2	Q1	Q1	Q1	Q1	Q1
D0	H12	P1	P1	P1	P1	P1
D1	H13	N2	N2	N2	N2	N2
D2	H14	N1	N1	N1	N1	N1
D3	J14	H2	H2	H2	H2	H2
D4	K14	M3	M3	M3	M3	M3
D5	K13	J2	J2	J2	J2	J2
D6	L14	L2	L2	L2	L2	L2
D7	K12	L3	L3	L3	L3	L3
D8	L13	F2	F2	F2	F2	F2
D9	N14	D1	D1	D1	D1	D1
D10	M12	E3	E3	E3	E3	E3
D11	N13	C1	C1	C1	C1	C1
D12	N12	G3	G3	G3	G3	G3
D13	P13	D2	D2	D2	D2	D2
D14	P12	K3	K3	K3	K3	K3
D15	M11	F3	F3	F3	F3	F3
D16	N11	J3	J3	J3	J3	J3
D17	N10	D3	D3	D3	D3	D3
D18	P11	C2	C2	C2	C2	C2
D19	P10	B1	B1	B1	B1	B1
D20	M9	A1	A1	A1	A1	A1
D21	N9	B2	B2	B2	B2	B2
D22	P9	A2	A2	A2	A2	A2
D23	N8	A4	A4	A4	A4	A4
D24	P7	A6	A6	A6	A6	A6

信号名	386DX	486SX	487SX	ODP	486DX	486DX2
D25	N6	B6	B6	B6	B6	B6
D26	P5	C7	C7	C7	C7	C7
D27	N5	C6	C6	C6	C6	C6
D28	M6	C8	C8	C8	C8	C8
D29	P4	A8	A8	A8	A8	A8
D30	P3	C9	C9	C9	C9	C9
D31	M5	B8	B8	B8	B8	B8
A20M#	–	D15	D15	D15	D15	D15
ADS#	E14	S17	S17	S17	S17	S17
AHOLD	–	A17	A17	A17	A17	A17
BE0#	E12	K15	K15	K15	K15	K15
BE1#	C13	J16	J16	J16	J16	J16
BE2#	B13	J15	J15	J15	J15	J15
BE3#	A13	F17	F17	F17	F17	F17
BLAST#	–	R16	R16	R16	R16	R16
BOFF#	–	D17	D17	D17	D17	D17
BRDY#	–	H15	H15	H15	H15	H15
BREQ#	–	Q15	Q15	Q15	Q15	Q15
BS8#	–	D16	D16	D16	D16	D16
BS16#	C14	C17	C17	C17	C17	C17
CLK	–	C3	C3	C3	C3	C3
D/C#	A11	M15	M15	M15	M15	M15
DP0	–	N3	N3	N3	N3	N3
DP1	–	F1	F1	F1	F1	F1
DP2	–	H3	H3	H3	H3	H3
DP3	–	A5	A5	A5	A5	A5
EADS#	–	B17	B17	B17	B17	B17
FERR#	–	–	A13	C14	C14	C14
FLUSH#	–	C15	C15	C15	C15	C15
HLDA	M14	P15	P15	P15	P15	P15
HOLD	D14	E15	E15	E15	E15	E15
IGNNE#	–	–	A15	A15	A15	A15
INTR	B7	A16	A16	A16	A16	A16
KEN#	–	F15	F15	F15	F15	F15
LOCK#	C10	N15	N15	N15	N15	N15
M/IO#	A12	N16	N16	N16	N16	N16
MP#	–	–	B14	B14	–	–
NMI	B8	A15	B15	B15	B15	B15
PCD	–	J17	J17	J17	J17	J17
PCHK#	–	Q17	Q17	Q17	Q17	Q17
PWT	–	L15	L15	L15	L15	L15
PLOCK#	–	Q16	Q16	Q16	Q16	Q16
RDY#	G13	F16	F16	F16	F16	F16
RESET	C9	C16	C16	C16	C16	C16
W/R#	B10	N17	N17	N17	N17	N17
TCK	–	–	–	–	A3*	A3
TD	–	–	–	–	A14*	A14
TD0	–	–	–	–	B16*	B16
TMS	–	–	–	–	B14*	B14
BUSY#	B9					
CLK2	F12					
ERROR#	A8					
NA#	D13					
PERQ	C8					

表1◆CPUの信号名とピンの対応表　＊印は50MHz版のみに存在する．VCC，VSSはi486系では同じ．N/Cもほとんど同じ．ただし，この表からは省いている．

は，制御信号の信号幅やタイミングの問題があり，倍クロック動作をするわけではありません．

例えば，PC-9801RA21の20MHz動作でi486DXを20MHz動作させようとすれば，40MHzのクロックを1/2に分周する必要があります．SS486では，ディップスイッチで，PCのクロックをそのままCPUに入れる場合と，1/2に分周してからCPUに入れる場合の2種類を設定できます．

i386以降の基本的なメモリやI/Oアクセスは非常に簡単な方法です．パイプライン方式を使うと少し複雑になるのですが，i486以降ではパイプライン方式のメモリアクセスはサポートされていません．そのため，基本の単純なメモリアクセスの方法を説明します．

クロックの立ち上りを基準にして，ADS#信号が開始信号で，CPUから出力されて周辺からRDY#信号が返ってくるまで待ちます．これが，1サイクルです．このた

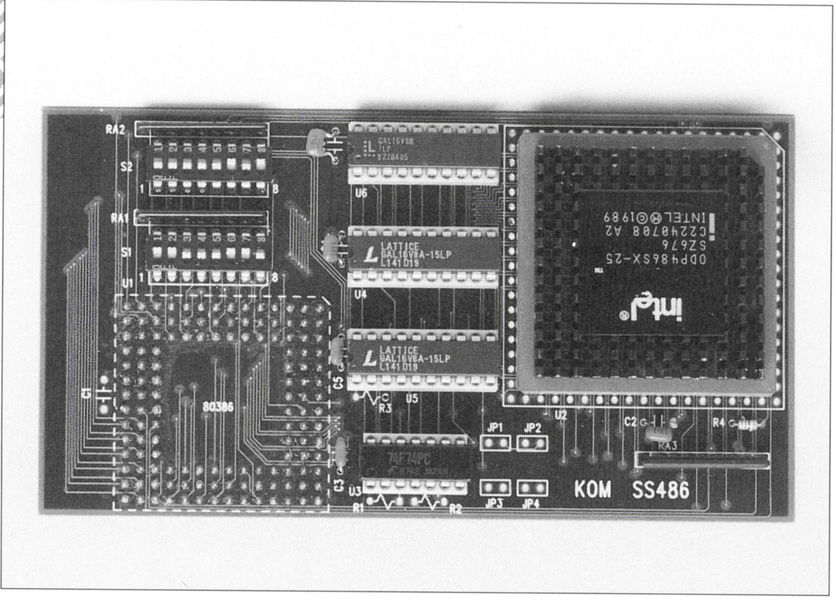


写真1◆SS486基板の部品面

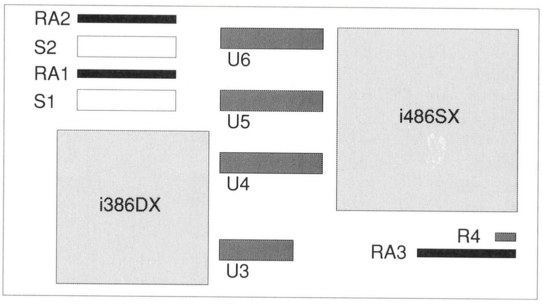


図1◆SS486を部品面から見た配置図

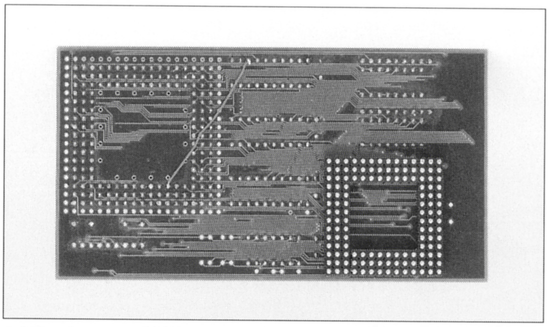


写真2◆SS486基板のハンダ面

め，信号のタイミングと信号の時間幅が合えば，CPUの動作クロックは2倍でも3倍でも4倍でも構わないことになります．CPUにとっては，ウェイトが増えただけです．したがって，処理速度が速くなって問題が出るのは，ソフトでタイミングを取っている場合で，別に対策が必要になります．

RA/DAのころのPCでは，周辺のI/Oを操作する場合に，LSI内の処理時間を確保するために“JMP $+2”が使用される場合が多かったようです．初期の設計段階で時間計算をして，安全をみて余分に入れているのでしょう．しかし，CPUを高速動作のアクセラレータに交換すると，その余裕がなくなります．そのため，起動時にFDDを認識しない，RS-232Cに文字落ちが発生する，CRT表示が異常になるなどの問題が発生します．

この症状をなくす方法としては，二つ考えられます．一つは，異常の発生するBIOS部分をキャッシュ禁止にする方法．もう一つは，異常の発生するI/Oアクセスがあった場合に，フラッシュ動作を行うという方法です．

キャッシュ禁止は簡単な方法で，これで異常が収まらないようなら，高速動作をあきらめるか，BIOS ROMを書き換えるしかないでしょう．アクセラレータを付けることで，PC本来の動作速度の2倍とか4倍になっているのですから，どこかに弱い部分が出てきて当然だと思います．

異常動作を起こした該当のI/Oアクセス時にフラッシュ動作をすると，正常動作になることがあります．これは内蔵キャッシュをフラッシュすると，CPUが新規に内部のバッファに1ライン分16バイトのデータを読み取るため，普通なら4回のデータバスの読み取り動作をしますから，この動作時間の分だけ周辺のICの時間の余裕が出て，安定動作をする可能性があるわけです．

接点1	0～32KB, 1MB～(1MB+64KB) 000000h ～ 003FFFh 100000h ～ 107FFFh
接点2	512KB～640KB 080000h ～ 09FFFFh
接点3	640KB～768KB 0A0000h ～ 0BFFFFh
接点4	768KB～896KB 0C0000h ～ 0DFFFFh
接点5	896KB～960KB 0E0000h ～ 0EFFFFh
接点6	960KB～1MB 0F0000h ～ 0FFFFFh

表2◆ キャッシュの設定
32Kバイト～512Kバイト(003FFFh～09FFFFh)は常にキャッシュ有効範囲.
14Mバイト以降(E00000h～FFFFFFh)はキャッシュ禁止範囲. この範囲をキャッシュONにすると, 最近流行のグラフィックアクセラレータが使用できないようです.

SS486の処理内容の説明

次に, SS486で行っている処理内容を, GALの論理の説明をすることで解説します.

●U4のGAL16V8の処理内容

キャッシュの範囲を指定します.

有効な24ビットのアドレスの上位を入力として, 範囲を決定しています. 特にテクニックは存在しません.

スイッチS1の該当する部分だけを**表2**に示します.

接点1は, HMAを使用する場合のためのものですが, ほとんどの場合ONにして問題はないと思います.

接点2は, I/Oバンクメモリボードを使用する場合に, その範囲をキャッシュするかどうかの設定です. S2の設定でI/Oバンクの制御があった場合に内蔵キャッシュをフラッシュする仕様にすれば, I/Oバンクメモリボードを使用しても, キャッシュ範囲に設定しても構いません.

接点3は, テキストVRAMやグラフィックVRAMの範囲です. OFFにして使用することをお勧めします. ONにした場合は表示が異常になる場合が多いです.

接点4は, ユーザ開放ROMの部分です. ハードウェアEMSの範囲でもあります. OFFで使用することをお勧めします.

接点5は, グラフィックVRAMとROM-BASICの部分です. OFFで使用することをお勧めします.

接点6は, ROM-BASICとBIOSの部分です. この部分のON/OFFが, 体感速度にかなり影響を与えます. ただし, ONにした場合には, 最悪誤動作をするかも知れません. 実機での動作確認の上, 判断してご使用ください.

接点1	RS-232C, HDD, FDDなど
接点2	グラフィックVRAM, プリンタ, 内蔵タイマ
接点3	テキストVRAM, ユーザ解放I/O, I/Oバンクメモリボードなど

表3◆ フラッシュの設定

元々はキャッシュのことなど考慮されていないi386DX機に, 内蔵キャッシュを持ったi486を使用するのですから, 単純にCPUの全メモリ空間をキャッシュすると異常が発生します. 例えば, 1文字表示するはずのCRT画面に, 同じ文字が2文字表示される場合があります. グラフィック画面に, ゴミのような余分な表示が出ることもありました.

i386DX搭載のPC98は, 16Mバイトまでしか管理していませんから, 24ビットのアドレスが有効です. その中で上位9ビットを使用して範囲を設定しています. それから, これはメモリアクセス時のみに有効になるようになっています. I/Oアクセスは, キャッシュの対象にはならない仕様にCPUがなっているのですが, 念のためです.

●U5のGAL16V8の処理内容

フラッシュの範囲を指定します(**表3**).

i486は, 内部キャッシュを8Kバイト持っています. この内部キャッシュにあるデータにヒットした場合には, 外部のメモリにアクセスせずにキャッシュ内のデータを高速に読み取るため, 外部メモリにアクセスする場合より高速動作ができます. そのキャッシュの内容と実際のメモリの内容に不一致が発生しないように, 必要に応じて外部からフラッシュ動作を行い, 内蔵キャッシュの内容をクリアします.

最初からi486を使用するPCなら, キャッシュをフラッシュする必要のある部分は分かっていますから, BIOS部分やハードウェアで, フラッシュ機能を動作させるようにできます. しかし, i386DX機などでは, 内蔵キャッシュのことはまったく考慮されていませんし, その必要もありませんでした.

そこで, 追加するユニットの部分でフラッシュの機能を実現しました. まず, 内蔵キャッシュとメモリの不一致が発生する可能性があるのはどのような場合なのかを

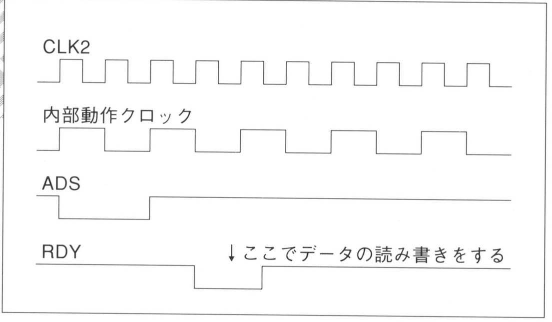


図2◆i386DXのノンパイプライン時のメモリアクセスウェイトなしの場合の動作

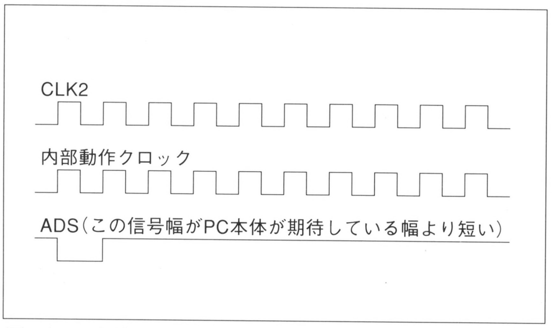


図3◆PC本体のクロックをi486に接続した場合の動作

考えてみます．

❶DMAや外部バスマスタが動作して，CPUが関知しない間にメモリを書き換えた場合
❷バンク切り替えやハードウェアEMSで実際のメモリが切り替わり，メモリの内容が変更になった場合
❸A20ラインの制御でのメモリアクセスの番地のずれが生じた場合

❶については，DMAが発生したら，必ずフラッシュする仕様にしています．厳密には，不要なフラッシュ動作をするため，効率の低下があります．例えば，DMAを用いたHDDの書き込み時には，メモリの内容が変化するわけではありませんから，フラッシュの必要はありません．ただし，この読み書きで区別するには，SS486基板の方の回路が大きくなってしまいます．したがって，回路の大きさと実用面での処理速度との兼ね合いで，現状の仕様にしました．

❷については，I/Oバンク方式のメモリボード，BIOSの一部，メモリウィンドウなどが相当します．内蔵キャッシュとの不整合を避けるには，キャッシュ範囲の設定でキャッシュ禁止にするか，バンクのI/Oアクセス時にフラッシュするかの二つの方法しかないと思います．

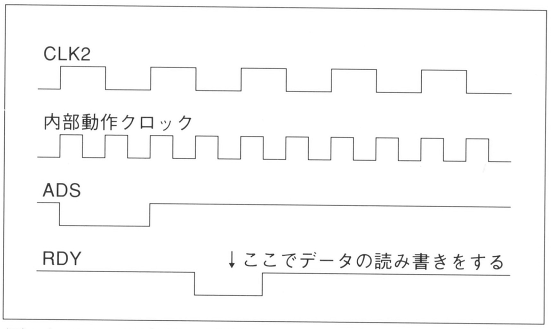


図4◆i486DX2内部で実行している処理

SS486では，どちらの方法でも選択できるようになっています．キャッシュ可能にして，I/Oでフラッシュする方式の方が体感速度は上だと思います．

❸については，i486以降では，A20を制御している情報をCPUに伝えるピンが存在しています．ただし，その制御はリアルモード時のみ必要なのです．ほとんどの場合，仮想86モードかプロテクトモードで動作しているため，A20ラインの制御情報を知ることは不要です．

S2-1，2，3以外にPC-98で未公開のバンク切り替えのポートなどのアクセスでは，フラッシュするようにしています．この部分は，PC-98の仕様に依存しています．どのI/Oでフラッシュするかで，ベンチマークテストの値に大きな影響を与えます．

●U6のGAL16V8の処理内容

ADS信号の補正とRDY信号の加工をしています．

ADS信号は，PC本体が必要としているタイミングに合わせるように補正しています．CPUが倍クロックで動作した場合には，信号幅が半分になり，PC本体の期待する信号幅になりません．

RDY信号の加工は，倍クロック動作をする場合に，PC本体からの信号では誤動作する場合があるので，信号幅などを加工しています．

まず，ADS信号の加工について詳しく説明します．

i386以降は，クロックの立ち上りが基準です．i386DXのノンパイプライン時のメモリアクセスウェイトなしの場合の動作は**図1**のようになります．

PC本体のクロックをi486に接続した場合，**図2**のようになります．この補正に必要なことはi486DXとi486DX2の関係で，i486DX2内部で実行している処理と同じことです(**図3**)．

CPU内部では，規定の時間以内に立ち下がり，時間

幅も規定通りになって当然です．ところが，外付け回路で同様の補正を行おうとすると，使用するICの遅延時間が問題になってきます．40MHzということは，一つのクロックが25ns(10^{-9}秒)ですから，普通のTTLやC-MOSでは，一つのゲートを通過する時間です．このために高速の素子が必要になってきます．また，ADSが出力されれば，単純に延ばすだけでは，外部とタイミングが合わない場合の補正も必要です．

この部分は，PC-98の仕様に依存していません．

また，RDY#信号は時間幅などを加工しないと3機種(RA2/RA21/DA)対応にはできませんでした．

i486の機能のうち SS486で使っていない機能

❶マルチプロセッサ機能

単独で使用するので不要です．

❷バースト転送

二次キャッシュのための機能なので未使用です．メモリのアクセスタイムに余裕があるなら，別にBRDY信号を作成することで，使用できるかも知れません．しかし，実際にはメモリにそれほどの余裕はないと思います．

高速の二次キャッシュが接続できるなら，さらに速度は向上するはずです．現状では，物理的な問題とコストパフォーマンスに問題があると思います．

❸バス・スヌーピング

ホールド時にDMAなどがメモリを書き換えた場合に，それをアドレスラインが入力となって，該当するキャッシュのラインだけをクリアする機能です．

現在はDMA動作時には，8Kバイトの内蔵キャッシュすべてをクリアしていますから，この機能が有効に使用できれば，体感速度が向上するはずです．

メモリ書き込み信号をCPUのピン以外から取れば，この機能を使用できる可能性があると思いますが，アクセラレータ装着時の手間を考えると標準採用できません．趣味のラベルならU5の論理を変更して，I/Oフラッシュをバンク切り替えを中心に書き換えて，PC本体から信号を取ればできると思います．ただし，この場合でもPC-98のハード仕様の問題があり，すべてのメモリ空間で使用できるかどうかは不明です．

Cx486SLC/DLCに換装した場合に，内蔵キャッシュをフラッシュさせる信号を拡張スロットから取る方法が紹介されていますが，内部増設メモリとメインメモリのアクセス時には，拡張スロットにはメモリアクセスの信号が出ない場合がありますから，これ以外からも信号を取るようにしないと100%安全とは言えません．

現在確認している 不具合

SS486の製作にあたり，NIFTY-Serve FDEVICEでベータテストを行わせていただきました．その結果，分かった不具合は以下の通りです．

❶RA2とRA21の一部で，電源ONで起動できない場合があります．もう1度電源をOFFにしてからONにすると，起動することが多いようです．これは，SS486にODPかi486DX2を搭載して，CPUが倍クロック動作をしているときに起こります．

PC本体クロックを分周したときに，PC本体の位相とずれる場合があるようです．これは，PC本体の周辺LSIの仕様に影響を受けます．DAではこの症状が発生する可能性は低いようです．

❷DAにアイ・オー・データ機器の内部増設メモリボードを付けていると，SS486の基板の大きさの制限で，SS486とメモリボードが接触することがあります．NIFTY-Serveでベータテストを行ったときには，メモリボードのコネクタのピンを切って，ぎりぎり入ったという報告もありました．

❸RA2を改造して20MHz動作をさせているPCでは，仮想86モードのドライバが起動した瞬間に暴走したという報告もありました．これは，RA2改が，ぎりぎりの所で動いているためだと思います．

❹RA21で，ODP-20を使ってCPUの動作周波数を40MHzで動作させた場合に，セラミックの表面温度が64℃で，メモリがパリティエラーを起こして停止したことがあったそうです．このODPにはヒートシンクを付けていなかったので，条件としては最悪だと思います．ファンで強制空冷させると，正常動作をしたようです．また，ディップスイッチの設定を標準から変更して，S2-5 ON，S2-6 OFFにした場合には，ファンなしでも安定動作をしたようです．このときのODP-20の表面温度は72℃で飽和したそうです．この症状から判断す

●テスト環境

NEC MS-DOS 5.0
Microsoft Windows Ver.3.1(MS版)

PC-9801DA(1.6MB+内部増設12MB)
BIOSはキャッシュ禁止
・条件DA1
DEVICE=SS486ON.SYS
DEVICE=HIMEM.SYS
DEVICE=RAMDISK.SYS 2048 1024 128
DEVICE=SMARTDRV.SYS 5120 2048
・条件DA2
DEVICE=SS486ON.SYS
DEVICE=HIMEM.SYS ←Windows3.1付属のもの
DEVICE=EMM386.EXE /F=D000 /P=156 /UMB /I=A600-A7FF /I=C000-CFFF /I=E900-F6FF /A=10 /VCPI
DEVICEHIGH A:¥DOS¥RAMDISK.SYS 2048 1024 128
DEVICEHIGH A:¥DOS¥SMARTDRV.SYS 5120 2048

PC-9801RA21(1.6MB+内部増設4MB+拡張スロット4MB)
PC-9801RA2(1.6MB+内部増設3MB+拡張スロット4MB)
BIOSはキャッシュ許可
・条件R1
DEVICE=SS486ON.SYS
DEVICE=HIMEM.SYS
DEVICE=RAMDISK.SYS 512 128
DEVICE=SMARTDRV.SYS 512
・条件R2
DEVICE=SS486ON.SYS
DEVICE=HIMEM.SYS
DEVICE=EMM386.EXE /F=D000 /P=32 /UMB /I=A600-A7FF /I=C000-CFFF /I=E900-F6FF /VCPI
DEVICEHIGH RAMDISK.SYS 512 128
DEVICEHIGH SMARTDRV.SYS 2048 128

●使用したソフト

測定項目A
CPUBENCH v0.980
Ratio to the first PC9801 : **A1**
Execute time : **A2** sec.

測定項目B
EMSBENCH /X V2.50
論理ページの物理ページへの割当は **B1** μ秒です.
ページマップのセーブ&リストアは **B2** μ秒です.
複数の論理/物理ページの割当は **B3** μ秒です.
部分ページマップの取得/設定は **B4** μ秒です.
EMSメモリアクセス速度(65536＊100バイト)は **B5** m秒です.
UMB(C000)アクセス速度(4096＊1600バイト)は **B6** m秒です.
=== XMS測定 ===
EMBの割り当て・解放時間は **B7** μ秒です.
EMB-EMB間の16K＊1000回データ移動は **B8** m秒です.
EMB-メイン間の16K＊1000回データ移動は **B9** m秒です.
メイン-メイン間の16K＊1000回データ移動は **B10** m秒です.
=== 参考データ ===
通常メモリアクセス速度(65536＊100バイト)は **B11** m秒です.
ROMアクセス速度(65536＊100バイト)は **B12** m秒です.
GVRAMアクセス速度(65536＊100バイト)は **B13** m秒です.
GVRAM裏アクセス速度(65536＊100バイト)は **B14** m秒です.

測定項目C
PFM486 Ver1.71
i486 Frequency = **C1** MHz
Ecache Read Time = **C2** ns/dword
Ecache Write Time = **C3** ns/word
Memory Read Time = **C4** ns/dword
Memory Write Time = **C5** ns/word

測定項目D
Norton SI V4.51
CPU スピード(CI), PC-9801VX41 対比: **D1**

測定項目E
ECI V1.32
80286(10M)比 **E1** 倍
80287(10M)比 **E2** 倍

表4◆ ベンチマークテストの環境と使用したソフト

ると，RA21では，外付けのADS補正回路ではタイミングがぎりぎりで動いているため，CPU自体のADSでは，遅れは少ないので安定度が高いということになります．

上記の症状の機種では無理ですが，DAやRA21では，ODP-33を使用して，ディップスイッチで，クロックを40MHzに設定すると，CPU内部が80MHz動作で動く場合があります．ベータテストの感触からは，確率は高いと感じました．

i486系は，パッケージの表面温度が80℃から85℃が動作保証上限温度です．SS486にODP-25を載せて小さなヒートシンクを付けた場合には，連続動作させてもヒートシンクの表面と外気温の差は40℃くらいです．強制空冷すれば，外気温との差はほとんどありません．ODP-33を使って，CPU内部80MHz動作のときもファンさえあれば，CPUの表面は冷えています．

❺SS486を使用すると，フリーソフトウェアのHSB Ver.2.9以降が暴走するようになりました．Ver.2.8とVer.2.6では動作確認がとれています．

その他では，PFM486の古いバージョンでは，SS486が暴走するらしいです．Ver1.71では，正常動作をします．

制御ソフトについて

制御ソフトには，SS486ON.SYSとSS486.COMがあります．

SS486ON.SYSは常駐0で，単純に内蔵キャッシュをONにするものです．

```
DEVICE=A:¥SS486¥SS486ON.SYS
```

をできるだけCONFIG.SYSの先頭の方に組み込んで下

項目	DA1	DA2	RA21 R1	RA21 R2	RA2 R1	RA2 R2
A1	82.26	80.34	49.71	49.71	35.98	35.61
A2	00.84	00.86	01.39	01.39	01.92	01.94
B1		46		83		107
B2		23		44		57
B3		28		53		73
B4		37		70		92
B5		350		490		730
B6		170		340		420
B7	29	33	53	58	69	74
B8	1540	1680	1090	1800	2000	2310
B9	1320	1460	1090	1510	2010	2300
B10	1180	1330	1240	1500	2150	2460
B11	340	350	480	480	730	730
B12	520	450	480	480	730	730
B13	3240	3260	5180	5180	5000	5000
B14	520	530	520	520	870	870
C1	78.95	78.95	39.37	39.37	32.02	32.02
C2	107	107	107	107	201	201
C3	106	106	106	106	200	200
C4	107	107	107	107	201	201
C5	107	106	106	106	200	200
D1	11.22	11.00	5.77	5.77	4.66	4.66
E1	4.85	4.68	4.55	4.44	3.02	2.92
E2	14.12	12.81	14.71	13.46	12.45	11.33

表5◆ベンチマークテストの結果1

さい．MELWAREを使用する場合には，その直後に置いてみてください．

SS486.COMの書式は，以下の通りです．

SS486	現在の状態を示します．
SS486　ON	キャッシュをONにします．
SS486　OFF	キャッシュをOFFにします．

キャッシュの変更は，DPMI上ではできません．もちろんWindows内でもできません．アイ・オー・データ機器のVMM386.SYSを用いた場合は，現在の状態も正しく表示されない場合があります．DPMIでは，内蔵キャッシュのON/OFFを指定するレジスタを保護しているので，簡単には操作できません．また，仮想86のドライバによっては，キャッシュのON/OFFの状態を正常に読み出せない場合があるようです．

IPL方式は，BIOS部分をキャッシュ許可にすると暴走するので，正式にはサポートしない予定です．

ベンチマークテスト

80MHzと40MHzでは，最速になるようにディップスイッチの設定を変更しています．**表4**にベンチマークの環境とソフトを．**表5，6**に結果を示します．

◆

WCLK2方式*とSS486を合体させれば，簡単にi486機用のODPの倍クロック化(4倍速)はできると思っています．ご要望があれば編集部気付でお知らせください．

DAで4倍速動作の時のWINDOSOCK Ver.1.3の値	
CPU	408から454
VIDEO	248
DISK	153

表6◆ベンチマークテストの結果2

◆

最後に，SS486キットが世に出るようになった途中で協力してくれた方々に感謝します．NIFTY-ServeのFDEVICEのSYSOPのUno様(SDI00443)は，ベータテスト参加者の募集を掲示を利用する事を許可してくださいました．また，ベータテスト参加者の方々も初期には，不良のこともあったのですが，快く協力してくださいました．

WCLK2方式として公開していた事で，知り合った雉井様(GGD01353)にも色々とアドバイスをもらいました．

最後になりましたが，本稿で紹介したSS486のキット(CPUを除く部品と内蔵キャッシュ制御ソフトをセットにしたもの)を下記にて委託販売しています．興味のある方は，問い合わせてみてください．

テクニカルサンヨー

☎06-644-0785　FAX06-644-0811

〈参考文献〉

1)『マイクロプロセッサ・ハンドブック(英語版)』，インテルジャパン(CQ出版扱い)

2)『インテル486™ DXマイクロプロセッサ・データブック』，インテルジャパン(CQ出版扱い)

3)『PC-9801シリーズ　テクニカルデータブック HARDWARE編』，アスキー

4)小高輝真，清水和文，速水祐，『PC-9801スーパーテクニック』，アスキー

5)小高輝真，『98の未公開機能を探る』，The BASIC'90年11月号

6)『GAL DATA BOOK』，Lattice Semiconductor Corporation

＊WCLK2方式とは，386系のCLK2を倍にする回路を外部につけて，倍速動作をさせる方法です．Cx486系CPUの倍速動作のために作って，名付けたものです．

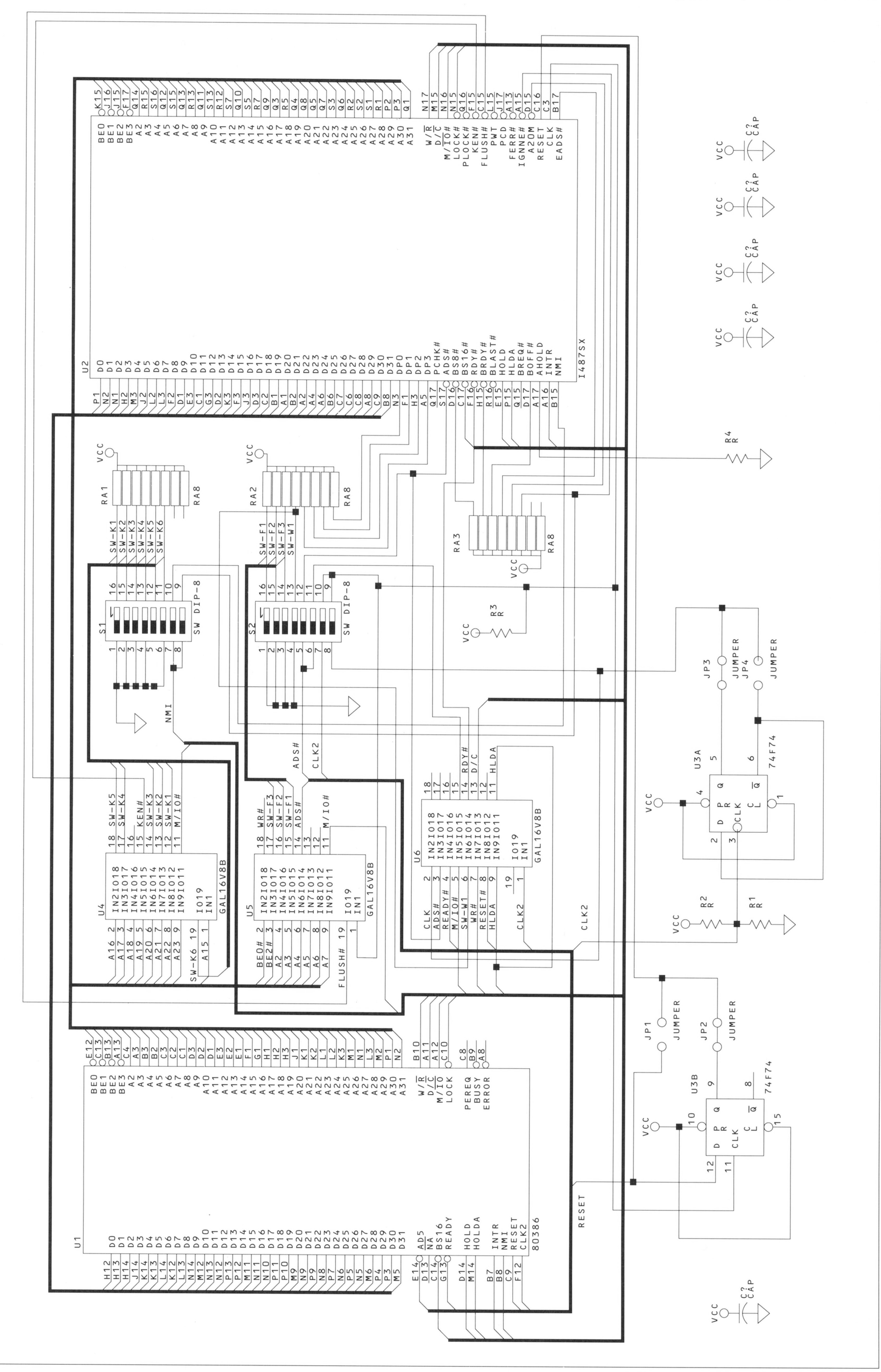


図5◆SS486の回路図

Advanced Assembler 386

マシンとの
より進化した
ディープな関係を築くために

不定期
連載 第6回

タスクスイッチ

■著者
速水　祐 *You HAYAMI*
新田　超二 *Chouji NITTA*
（ZOBplus）

SERVICE

はじめに

今回は，ページングを扱う予定でしたが，仮想86で動作するプログラムはかなり複雑なものがあり，今までの連載で解説した内容では不足です．そこで，今回から数回連続で仮想86モニタの制作のための解説をはじめることにし，その中でタスクスイッチ，割り込み，ページングの説明を行うことにしました．

最初である今回はタスクスイッチです．実例プログラムを作成していく過程において，プロテクトモードにおけるものより，仮想86モードにおけるページングの方が実動作として興味深いものがあります．プロテクトモードから仮想86モードへ移行するためには，通常はタスクスイッチが使用されます．よって，まずプロテクトモードにおける一般的なタスクスイッチの解説と実例を紹介し，仮想86モードへ移行するプログラムを紹介することにします．

タスクスイッチ

80286以降の80x86CPUは，直接タスクを切り替えることができます．8086ではタスク切り替え（タスクスイッチ）はソフトウェアのみで実現されており，その処理の複雑さと必要なCPUクロックは大きなものです．さらに保護機能も弱いため，実用的なマルチタスクの環境を作り出すのには非常にむずかしいものがありました．i386/i486では80286の16ビット処理を発展させた32ビット処理におけるタスクスイッチが可能となり，高度なマルチタスクの環境を実現することができます．

タスクスイッチはプロシージャ呼び出しに似ていますが，プロシージャ呼び出しがCS（コードセグメント）とEIP（インストラクションポインタ）しかセーブせず，それ以外のレジスタはPUSH命令を使って明示的にセーブしなければならないのに対し，CPUによるタスクスイッチは自動的に多くのプロセッサステート情報をセーブすることができ，切り替えた新タスクにおけるプロセッサステート情報がCPUにリストアされます．

したがって，タスクスイッチ機能により制御は完全に新しい環境に切り替えることが可能であり，マルチ

タスク環境においては必須の機能といえます．ただし，タスクスイッチによるプロセッサステート情報はそれぞれのタスクが所有するタスクステートセグメント(以下TSS)と呼ばれるメモリ内のデータエリアに格納されるため，スタックにデータがセーブされるプロシージャと異なりリエントラントではありません．

TSSとそのエリアを表すTSSディスクリプタ，現在のタスクのTSSディスクリプタを指すタスクレジスタ，そしてタスクゲートディスクリプタがタスクスイッチの制御を行う重要なシステムデータ構造体となります．これらのシステム構造体/レジスタについて解説していきます．

TSS(タスクステートセグメント)

TSSは一つのタスクあたり一つ必要とし，マルチタスクのときは複数必要なことになります．シングルタスクのプログラムにおいても，特権レベル移行などの処理が伴うときには確保する必要があります．TSSにはタスクを管理する情報が納められており，**図1**のようなフォーマットになっています．アミ部分は初期設定のときのみ変更され，それ以降変更が行われない静的フィールドで，それ以外はタスク切り替えの度に更新される動的フィールドです．

◆静的フィールド

● オフセット04H～19H

特権レベル0，1，2のスタックセレクタとスタックポインタ値であり，特権レベル遷移の際にこの値がスタックポインタとして利用されます．

● オフセット1CH

CR3システムレジスタが格納されており，ページテーブル(CR3はページディレクトリの先頭アドレスを指す)をタスクにより切り替えて使用できます．

● オフセット60H

各タスクのLDT(ローカルディスクリプタテーブル)セレクタであり，それぞれのタスクの異なるLDTを用意することにより独立したセグメント管理が可能となります．

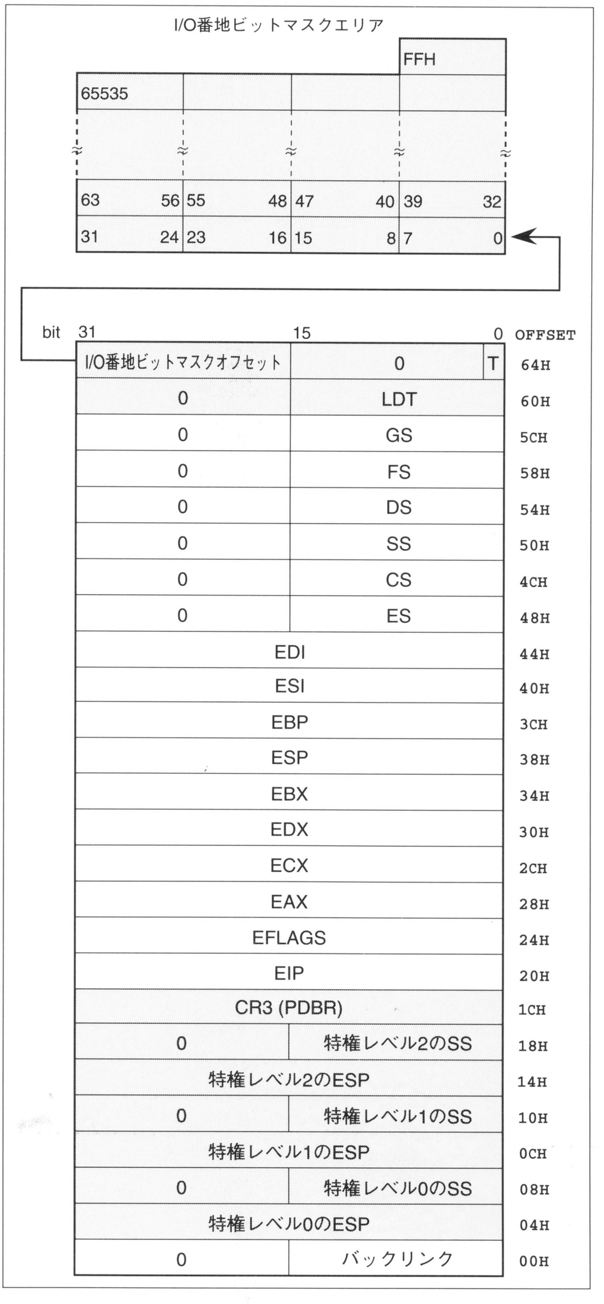
● オフセット64H

Tビット(デバッグトラップビット)．このビットがセットされているとタスク切り替えの際，デバッグ例外が発生します．

● オフセット66H

I/Oベースマップフィールドには I/O許可ビットマッ

▶図1●32ビットタスクステートセグメント



プのTSS内における開始オフセットが格納されています.

● オフセット68H以降

I/O許可ビットマップフィールドが，TSSのリミットまで格納されています.

◆動的フィールド

● オフセット20H，4CH

タスクスイッチが行われると，EIP，CSフィールドに格納された値から実行が開始されます．旧タスクのTSSのEIP，CSフィールドには，タスクスイッチ命令の次の命令ポインタが格納されています.

● オフセット24H～5DH

フラグレジスタ(EFLAGS)，汎用レジスタ(EAX，ECX，EDX，EBX，ESP，ESI，EDI)とセレクタ(ES，CS，SS，DS，FS，GS)のフィールドで，タスク切り替えの際CPUレジスタにセットされます.

セレクタは16ビット長なので32ビットのうち上位16ビットは0が格納されています.

● オフセット00H

バックリンクセレクタフィールドは前に実行されていたタスクのTSSセレクタの値が保存されており，フラグレジスタのNTフラグが1のときのIRET命令はこの値を使ってタスクスイッチを行います.

TSSのサイズはI/O許可ビットマップがない場合は，68H(104)バイトとなります．これ以下のサイズのTSSを持つタスクへ切り替えようとすると例外が発生します．TSSの占めるメモリエリアは，タスク切り替えによりページテーブルが変更される可能性があるため，ページ境界にまたがると修復不能エラーが発生することがあります.

TSSディスクリプタ

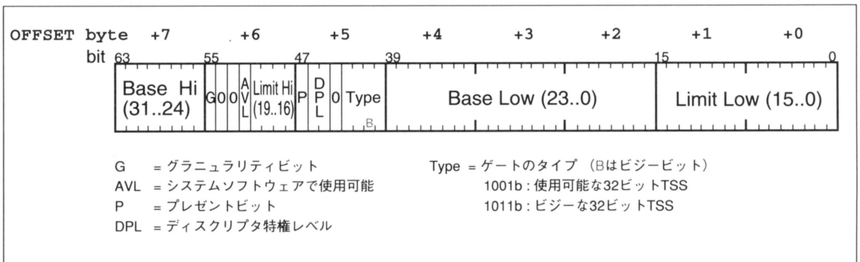
TSSディスクリプタは，TSSのメモリエリアを表すシステムディスクリプタでGDT(グローバルディスクリプタテーブル)にのみ納めることができます(**図2**)．ページアドレスはTSSの先頭アドレス，リミットはTSSサイズ-1の値が格納されるので，TSSの最小サイズ(104)-1＝103未満の値を入れることはできません．タイプフィールドは89Hとなり，そのタスクがビジー状態のときタイプフィールド内のビジービットがセットされ，タイプフィールドの値は8BHに書き変わります．ビジービットがセットされているタスクへは，IRET命令以外でスイッチすることはできません．この機能により，タスクが再帰的に呼び出されることを防止しています.

DPL(特権レベル)は通常0にセットされるので特権レベル0のみからこのディスクリプタを利用したタスク切り替えが可能となりますが，セグメントレジスタによる指定は不可能のため，直接TSSエリアをリード/ライトすることはできません．そこで他のTSSエリアを含むデータセグメントディスクリプタ(エイリアス)を設定して，これを通してアクセスすることになります.

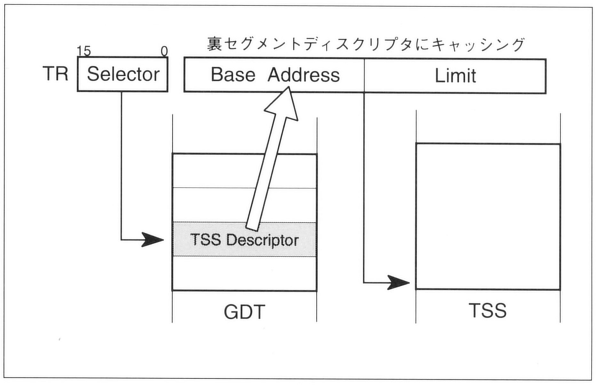
タスクレジスタ

タスクレジスタは，現在実行されているタスクのTSSディスクリプタを指すセレクタを格納します(**図3**)．LTR命令で意図的に設定するか，タスクスイッチの際自動的に設定されます．システムプログラムの初期設定においては，まだタスクスイッチは1度も発生し

▶図2●TSSディスクリプタ



▶図3●タスクレジスタとTSSディスクリプタ，TSSの関係



ていないわけですからLTR命令で設定をする必要があります．しかし，LTR命令によりTSSディスクリプタのセレクタがセットされても，タスクスイッチが行われるわけではありません．

特権レベル間の移行においてスタックポインタが変更されます．この値はTSSに格納されていますので，特権レベル間の移行のような動作をするプログラムにおいては，それがシングルタスクでもタスクレジスタを設定する必要があります．

タスクレジスタには，外からアクセスできない裏セグメントディスクリプタが備わっています．タスクスイッチの際，タスクレジスタが指すTSSディスクリプタの内容が格納された裏セグメントディスクリプタにより，高速にタスクプロセッサステート情報の退避等を行うことができます．

タスクゲートディスクリプタ

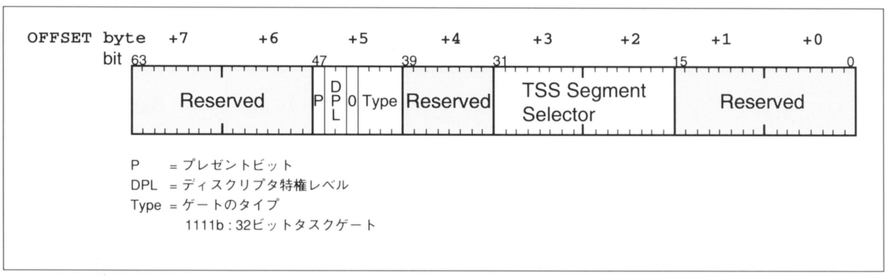
TSSディスクリプタのDPLより，低い特権レベルにおけるタスクスイッチが行えるように，タスクゲートディスクリプタがあります(**図4**)．低い特権レベルから高い特権レベルのプロシージャを呼ぶためにコールゲートディスクリプタが用意されているように，タスクゲートディスクリプタは，タスクスイッチにおけるコールゲートディスクリプタと同様な働きをします．ただし，コールゲートディスクリプタと異なり，タスクゲートディスクリプタはオフセットフィールドの値は使用せず，切り替え先のTSSディスクリプタを指すセレクタの内容がセレクタフィールドに格納されています．このセレクタフィールドのRPLとセレクタが指すTSSディスクリプタのDPLの内容は使用されず，特権レベルの判断はタスクゲートディスクリプタのDPLによってのみ行われます．(MAX(CPL，RPL)≦DPL)

タスクゲートディスクリプタはGDTだけではなく，各タスクのLDT内におくことができるため，特権レベルの低いタスクでも特権レベル0のTSSを使ったタスクスイッチを行うことができます．また，割り込みディスクリプタテーブルに存在することもでき，割り込みに伴うタスクスイッチが可能です．

タスクスイッチング

次に実際のタスクスイッチの動作の流れと，その実例を示します．タスクスイッチの方法には以下のよう

▶図4●タスクゲートディスクリプタ



なものがあり，ターゲットセレクタがTSSディスクリプタかタスクゲートディスクリプタであれば通常のJMP，CALL，IRET命令で実行できます．

- TSSディスクリプタを指すセレクタによるFAR JMP
- TSSディスクリプタを指すセレクタによるFAR CALL
- タスクゲートディスクリプタを指すセレクタによるFAR JMP
- タスクゲートディスクリプタを指すセレクタによるFAR CALL
- フラグのNTビットがセットされているときのIRET
- 割り込みベクタがタスクゲートディスクリプタのときの割り込み

◆TSSディスクリプタを指すセレクタによるFAR JMP（リスト1）

```
@jmpf   TSS2_SEL,0
```

命令でタスク切り替えが行われます．@jmpfはTT386.INC(**プログラムリスト2**)内で宣言されているマクロでセレクタによるFAR JMP(66H)にコード変換されます．TSS2_SELがタスク2のTSSディスクリプタを指すセレクタで，オフセット0は使用されません．

タスクスイッチの命令が実行されると，次のようにCPU内での動作が行われます．

❶ CPUは新TSSディスクリプタの内容から特権レベルチェック，存在チェック，リミットチェックを行います．
❷ いままでのタスクのプロセッサステート情報を旧TSSに格納します．TSSはタスクレジスタのセレクタで指定されますので，**リスト1**ではLTR命令によってTSSディスクリプタを示すタスクレジスタを事前にセットしています．
❸ タスクレジスタに新TSSディスクリプタを指すセレクタがロードされます．このとき新TSSディスクリプタ内のビジービットがチェックされ，ビジー状態のタスクならば例外が発生します．
❹ 新TSSからタスクプロセッサステート情報をCPU内にロードします．新TSSディスクリプタ内のビジービットがセットされ，旧TSSディスクリプタのビジービットがクリアされます．このとき，CR0内のTSビットもセットされ，数値演算コプロセッサのタスク間の使用調整に利用されます．
❺ ❹の動作時点ではセレクタのロードは行われますが，裏セグメントディスクリプタのロードは行われていません．LDTRがロードされLDTRの裏セグメントディスクリプタがセットされた後，他のセレクタの裏セグメントディスクリプタのセットが行われます．

ひとつのFAR JMP命令だけで❶～❺の動作が行われ，高速で信頼性の高いタスクスイッチが実現できます．しかし，これだけ大きな処理ですのでCPUタイムは，かなり多く消費されi386DXでは300クロックを必要とします．通常のFAR JMPは30クロック程度なのでタスクプロセッサステート情報を大幅に入れ替える必要がない場合は，ソフトウェアによるタスクスイッチの方が高速なこともあります．

リスト1では，スクリーン左上端に文字‘A’を表示した後，タスクスイッチを実行してタスク2のTSSにすでに格納されているEAX(＝‘B’)，EDI(＝160＊10)を利用して画面10行目の左端に文字‘B’を表示します．その後FAR JMPによるタスク1へのタスクスイッチを行います．タスク1のTSSには，タスク1におけるレジスタの内容や命令ポインタが格納されているので，それがロードされてFAR JMPの次の命令が実行され，さきほどの文字の右側にもう1文字‘A’が表示されます．

◆TSSディスクリプタを指すセレクタによるFAR CALL(リスト2)

FAR JMPの代わりに

```
@callf   TSS2_SEL,0
```

のFAR CALLになっています．FAR JMPとの動作の違いは，タスク2においてIRET命令でタスク1に復帰できることでです．タスクスイッチの際，旧タスクのTSSディスクリプタ内のビジービットはFAR JMPのときと異なり，セットされた状態のまま変化しません(**表1**)．

そして，新タスクのフラグのNTビットがセットされ，新タスクのTSSのバックリンクセレクタフィールドに旧タスクのTSSディスクリプタを指すセレクタが格納されます．

▶リスト1

```
;
;---------- Task Test program 1
;	tt386とリンクして実行ファイルを作成する

	.386P

	INCLUDE tt386.inc

_PTEXT	SEGMENT PARA PUBLIC USE32 'PCODE'
		ASSUME cs:_PTEXT,ds:DGROUP
ttest:
	;----- タスク1
	mov	ax,LDT1_SEL
	lldt	ax
	mov	ax,TVRAM1_SEL
	mov	es,ax			; テキストVRAMセグメントセット
	mov	edi,0			; OFFSET 0 左上端
	mov	ax,'A'
	mov	es:[edi],ax		; 文字'A'をライト
	add	edi,2
	@jmpf	TSS2_SEL,0		; タスクスイッチ
	mov	es:[edi],ax		; 文字'A'をライト
	jmp	RetReal

task2:
	;----- タスク2
	mov	dx,TVRAM2_SEL
	mov	es,dx			; テキストVRAMセグメントセット
	mov	es:[edi],ax		; 文字'B'をライト
	add	edi,2
	@jmpf	TSS1_SEL,0

_PTEXT	ENDS
	END
```

▶リスト2

```
;
;---------- Task Test program 2
;
	.386P

	INCLUDE tt386.inc

_PTEXT	SEGMENT PARA PUBLIC USE32 'PCODE'
		ASSUME cs:_PTEXT,ds:DGROUP
ttest:
	;----- タスク1
	mov	ax,LDT1_SEL
	lldt	ax
	mov	ax,TVRAM1_SEL
	mov	es,ax			; テキストVRAMセグメントセット
	mov	edi,0			; OFFSET 0 左上端
	mov	ax,'C'
	mov	es:[edi],ax		; 文字'A'をライト
	add	edi,2
	@callf	TSS2_SEL,0
	mov	es:[edi],ax		; 文字'A'をライト
	jmp	RetReal

task2:
	;----- タスク2
	mov	dx,TVRAM2_SEL
	mov	es,dx			; テキストVRAMセグメントセット
	mov	es:[edi],ax		; 文字'B'をライト
	add	edi,2
	iret

_PTEXT	ENDS
	END
```

▶リスト3

```
;
;---------- Task Test program 3
;
	.386P

	INCLUDE tt386.inc

_PTEXT	SEGMENT PARA PUBLIC USE32 'PCODE'
		ASSUME cs:_PTEXT,ds:DGROUP
ttest:
	;----- タスク1
	mov	ax,LDT1_SEL
	lldt	ax
	@jmpf	TSS2_SEL,0
	jmp	RetReal

task2:
	;----- タスク2
	xor	eax,eax
	mov	al,[gdt_tss1].access
	mov	edi,0
	call	HexPut
	pushfd
	pop	eax
	mov	edi,160
	call	HexPut
	@jmpf	TSS1_SEL,0

_PTEXT	ENDS
	END
```

▶リスト4

```
;
;---------- Task Test program 4
;
	.386P

	INCLUDE tt386.inc

_PTEXT	SEGMENT PARA PUBLIC USE32 'PCODE'
		ASSUME cs:_PTEXT,ds:DGROUP
ttest:
	;----- タスク1
	mov	ax,LDT1_SEL
	lldt	ax
	@callf	TSS2_SEL,0
	jmp	RetReal

task2:
	;----- タスク2
	xor	eax,eax
	mov	al,[gdt_tss1].access
	mov	edi,0
	call	HexPut
	pushfd
	pop	eax
	mov	edi,160
	call	HexPut
	iret

_PTEXT	ENDS
	END
```

新タスクにおいてIRET命令が実行されると通常のCS，EIP，EFLAGSをPOPする動作と異なり，バックリンクセレクタフィールドの値を使って，タスク復帰のためのタスクスイッチが行われます．

◆ビジービット，NTフラグの変化のテスト（リスト3，リスト4）

リスト3，リスト4は，FAR JMP，FAR CALLにおけるタスクスイッチの際の新旧タスクのTSSディスクリプタ内のタイプフィールドの状態とフラグの値を表示するものです．

リスト3を実行すると画面左上に，

00000089
00000206

と表示され，この値により，旧タスクのタイプフィールドのビジービットがクリアされ，新タスクのフラグのNTビットがクリアされていることを確認できます．

リスト4を実行すると画面左上に，

0000008B
00004206

と表示され，この値により，旧タスクのタイプフィールドのビジービットがセットされ，新タスクのフラグのNTビットがセットされており，FAR JMPとFAR CALLによるフラグの変化の違いを確認できます．

◆タスクゲートディスクリプタを指すセレクタによるFAR JMPとFAR CALL（リスト5，リスト6）

タスクゲートディスクリプタを使った間接的なタスクスイッチです．特権レベルのチェックがTSSディスクリプタに対して行われるのではなく，タスクゲートディスクリプタのDPLとのチェックが行われます．

リスト1		リスト5
@jmpf TSS2_SEL,0	→	@jmpf TASKGATE_SEL,0
リスト2		**リスト6**
@callf TSS2_SEL,0	→	@callf TASKGATE_SEL,0

▶リスト5

```
;
;---------- Task Test program 5
;

        .386P

        INCLUDE tt386.inc

_PTEXT  SEGMENT PARA PUBLIC USE32 'PCODE'
                ASSUME cs:_PTEXT,ds:DGROUP
ttest:
        ;----- タスク1
        mov     ax,LDT1_SEL
        lldt    ax
        mov     ax,TVRAM1_SEL
        mov     es,ax                           ; テキストVRAMセグメントセット
        mov     edi,0                           ; OFFSET 0 左上端
        mov     ax,'A'
        mov     es:[edi],ax                     ; 文字'A'をライト
        add     edi,2
        @jmpf   TASKGATE_SEL,0                  ; タスクゲートによるタスクスイッチ
        mov     es:[edi],ax                     ; 文字'A'をライト
        jmp     RetReal

task2:
        ;----- タスク2
        mov     dx,TVRAM2_SEL
        mov     es,dx                           ; テキストVRAMセグメントセット
        mov     es:[edi],ax                     ; 文字'B'をライト
        add     edi,2
        @jmpf   TSS1_SEL,0

_PTEXT  ENDS
        END
```

▶リスト6

```
;
;---------- Task Test program 6
;
        .386P

        INCLUDE tt386.inc

_PTEXT  SEGMENT PARA PUBLIC USE32 'PCODE'
                ASSUME cs:_PTEXT,ds:DGROUP
ttest:
        ;----- タスク1
        mov     ax,LDT1_SEL
        lldt    ax
        mov     ax,TVRAM1_SEL
        mov     es,ax                           ; テキストVRAMセグメントセット
        mov     edi,0                           ; OFFSET 0 左上端
        mov     ax,'A'
        mov     es:[edi],ax                     ; 文字'A'をライト
        add     edi,2
        @callf  TASKGATE_SEL,0                  ; タスクゲートによるタスクスイッチ
        mov     es:[edi],ax                     ; 文字'A'をライト
        jmp     RetReal

task2:
        ;----- タスク1
        mov     dx,TVRAM2_SEL
        mov     es,dx                           ; テキストVRAMセグメントセット
        mov     es:[edi],ax                     ; 文字'B'をライト
        add     edi,2
        iret

_PTEXT  ENDS
        END
```

リスト1，2と比べてセレクタの指定のみが変わっています．

間接的なタスクスイッチですので，TSSディスクリプタ経由のタスクスイッチより9クロック余計に実行時間がかかります．

◆仮想86モードへチェンジ(リスト7)

最後にタスクスイッチを利用した仮想86モードへのスイッチを示します．

TT386.ASM内のTss2Initプロシージャが**リスト8**に入れ代わります．TSSに設定するセレクタがリアルモードと同様なセグメントとなり，フラグがIOPL＝3,仮想86フラグVM＝1にセットされることが異なります．タスクスイッチが行われると，ここで設定されたプロセッサステート情報がCPUに転送され，初期設定処理なしで仮想86モードが実行できます．特権レベル0においてスタックに退避させているフラグのVMビットがセットされているとIRET命令で仮想86モードへ移行できますが，タスクスイッチによるモード移行の方が初期設定のことを考え合わせるとベターな方法です．

リスト7のプログラムは仮想86モードへ移行するとそのまま無限ループに入ります．仮想86モードでは単純なタスクスイッチではプロテクトモードへ戻ることはできないためです．仮想86におけるプログラム実例は次回以降ということで．

おわりに

最後に仮想86モードへ移行できましたので，次回はこのモードにおける割り込みについての解説を行う予定です．

連載第1回で紹介したZOB Station BBSは会員も大幅に増えたため，8回線に増強し，それに伴い電話番号の変更がなされています．ホストマシンにIBM-AT互換機i486DX2 66MHzに16550A×4×2のシリアルボードとハードディスクキャッシュ16Mバイトを搭載して，外部8回線で最高速の転送スピード(最高3,000cps)を実現しています．プログラム作成に関する活発な情報交換も行われていますので，ぜひアクセスしてみてくだ

▶リスト7

```
;
;---------- Task Test program 7
;
;       Tss2InitをLIST 8と入れ替えた
;       tt386とリンクして実行ファイルを作成する
;
        .386P

        INCLUDE tt386.inc

_PTEXT  SEGMENT PARA PUBLIC USE32 'PCODE'
                ASSUME cs:_PTEXT,ds:DGROUP
ttest:
        ;----- タスク1
        mov     ax,LDT1_SEL
        lldt    ax
        mov     ax,TVRAM1_SEL
        mov     es,ax                   ; テキストVRAMセグメントセット
        mov     edi,0                   ; OFFSET 0 左上端
        mov     ax,'A'
        mov     es:[edi],ax             ; 文字'A'をライト
        add     edi,2
        @jmpf   TSS2_SEL,0
_PTEXT  ENDS

_TEXT   SEGMENT PARA PUBLIC USE16 'CODE'
        ASSUME  cs:_TEXT,ds:DGROUP
;
;---------- 仮想86 タスク
;
task2   PROC
        mov     dx,0A000h               ; 仮想86モードにおける
        mov     es,dx                   ; テキストVRAMセグメントセット
        mov     es:[di],ax              ; 文字'B'をライト
        jmp     $                       ; 無限ループ
task2   ENDP
_TEXT   ENDS

        END
```

▶リスト8

```
; TT386.ASM内のTss2Initをこのリストと入れ替えて
; 仮想86モード移行のテストプログラムLIST 7とリンクします

;---------- TSS2 Intialize

Tss2Init PROC NEAR
        mov     bx,OFFSET DGROUP:tsst2
        mov     ax,cs
        mov     [bx].TSS.tss_cs,cs
        mov     ax,DGROUP
        mov     [bx].TSS.tss_ds,ax
        mov     [bx].TSS.tss_ss,ax
        mov     [bx].TSS.tss_fs,ax
        mov     [bx].TSS.tss_gs,ax
        mov     [bx].TSS.tss_es,ax
        mov     [bx].TSS.tss_eip,OFFSET _TEXT:task2
        mov     [bx].TSS.tss_esp,OFFSET DGROUP:stack_top2
        mov     [bx].TSS.tss_eax,'B'
        mov     [bx].TSS.tss_edi,160 * 10
        mov     [bx].TSS.tss_eflags,023202h     ; VM = 1, IOPL = 3
        mov     [bx].TSS.tss_cr3,0
        ret
Tss2Init ENDP
```

さい.

ZOB Station BBS

14,400/9,600bps V.32bis/V.32 7回線+
38,400 V.110 1回線(V.32と兼用)
(全回線無料)
会員数　2,300名
アクセス数　650以上(1日平均)
通信条件　パリティなし，データ8ビット
ストップビット1ビット
電話番号　0466-80-1045　(V.32bis/V.32)
0466-80-1048　(V.110)

今回のプログラムリストや，今までの連載(VCPIの回を除く)で掲載したものは，i386のプロテクト命令を多く使用しているため仮想86EMSエミュレータ(EMM386.EXE，MELEMM.386，VMM386.SYSなど)上では動作しませんので注意してください.

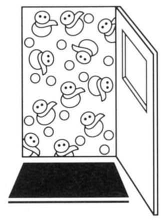
<参考文献>

1)『386DXマイクロプロセッサ・プログラマーズ・リファレンスマニュアル第2版』，インテルジャパン(CQ出版扱い)

2)『386DX Microprocessor Programmer's Reference Manual '90』，Intel

3) Robert L.Hummel，『PROGRAMMER'S TECHNICAL REFERENCE The Processor and Coprocessor』，ZD Press
上の訳本が最近発刊されました.

4) Robert L.Hummel，槌田浩一訳，『80x86/80x87ファミリテクニカルハンドブック』，技術評論社



・・・プログラムリスト1　TT386.ASM・・

```
;-----------------------------------------------------------------------------
;        386 Task Switch TEST Program
;
;                                  ZOBplus Hayami
;                                  1993 10/1
;-----------------------------------------------------------------------------

        .386P

        INCLUDE tt386.inc

;
;---------- 16 bits data segment --------------------------------------------
;
_DATA   SEGMENT PARA PUBLIC USE16 'DATA'
;---------- Global Descriptor ------------------
gdt             LABEL QWORD
gdt_null        DESC <0,0,0,0,0,0>
gdt_rcode       DESC <0FFFFh,0,0,9ah,0,0>        ; Access = 9ah
                                                 ; dpl=0,code,non conf,read ok
                                                 ; Attribute = 0h
                                                 ; byte,use16
                                                 ; Limit=FFFFh 64k
gdt_rdata       DESC <0FFFFh,0,0,92h,0,0>        ; Access = 92h
                                                 ; dpl=0,data,up,write ok
                                                 ; Attribute=0h
                                                 ; byte,use16
                                                 ; Limit=FFFFh 64k
gdt_pcode       DESC <0FFFFh,0,0,09ah,0cfh,0>    ; Access = 9ah
                                                 ; dpl=0,code,non conf,read ok
                                                 ; Attribute = 0Ch
                                                 ; page,use32
                                                 ; Limit=FFFFFh 4G
gdt_pdata       DESC <0FFFFh,0,0,92h,0cfh,0>     ; Access = 92h
                                                 ; dpl=0,data,up,write ok
                                                 ; Attribute=0Ch
                                                 ; page,use32
                                                 ; Limit=FFFFFh 4G
gdt_vram        DESC <02000h-1,0,0ah,92h,0,0>    ; Access = 92h
                                                 ; dpl=0,data,up,write ok
                                                 ; Attribute=0h
                                                 ; byte,use16
                                                 ; Limit=2000h 8k
gdt_tss1        DESC <TSSLIMIT,0,0,89h,0,0>      ; Access = 89h
                                                 ; dpl=0,type = 09h
                                                 ; Limit=TSSLIMIT
gdt_tss2        DESC <TSSLIMIT,0,0,89h,0,0>      ; Access = 89h
                                                 ; dpl=0,type = 09h
                                                 ; Limit=TSSLIMIT
gdt_ldt1        DESC <8-1,0,0,82h,0,0>           ; Access = 82h
                                                 ; dpl=0,type = 02h
                                                 ; Limit=8
gdt_ldt2        DESC <8-1,0,0,82h,0,0>           ; Access = 82h
                                                 ; dpl=0,type = 02h
                                                 ; Limit=8
gdt_taskgate    GDESC <0,TSS2_SEL,0,85h,0>       ; Task Gate
                                                 ; type = 05h
```

```
GDTLIMIT        equ     $-gdt-1

ldt1            LABEL QWORD
ldt1_vram       DESC <02000h-1,0,0ah,92h,0,0>   ; Access = 92h
                                                ; dpl=0,data,up,write ok
                                                ; Attribute=0h
                                                ; byte,use16
                                                ; Limit=2000h 8k
ldt2            LABEL QWORD
ldt2_vram       DESC <02000h-1,0,0ah,92h,0,0>   ; Access = 92h
                                                ; dpl=0,data,up,write ok
                                                ; Attribute=0h
                                                ; byte,use16
                                                ; Limit=2000h 8k

;---------- Selector Define ---------------------
RCODE_SEL       equ     gdt_rcode - gdt         ; 8h
RDATA_SEL       equ     gdt_rdata - gdt         ; 10h
PCODE_SEL       equ     gdt_pcode - gdt         ; 18h
PDATA_SEL       equ     gdt_pdata - gdt         ; 20h
TVRAM_SEL       equ     gdt_vram - gdt          ; 28h
TSS1_SEL        equ     gdt_tss1 - gdt          ; 30h
TSS2_SEL        equ     gdt_tss2 - gdt          ; 38h
LDT1_SEL        equ     gdt_ldt1 - gdt          ; 40h
LDT2_SEL        equ     gdt_ldt2 - gdt          ; 48h
TASKGATE_SEL    equ     gdt_taskgate - gdt      ; 50h
TVRAM1_SEL      equ     ldt1_vram - ldt1 + 04h  ; 04h
TVRAM2_SEL      equ     ldt2_vram - ldt2 + 04h  ; 04h

;---------- GDT pointer ------------------------
gdt_ptr         LABEL   FWORD
gdt_limit       dw      GDTLIMIT
gdt_base        dd      ?

oldgdt_ptr      LABEL   FWORD
oldgdt_limit    dw      ?
oldgdt_base     dd      ?

;---------- Work area --------------------------
old_sp          dw      ?                       ; Old stack pointer
old_ss          dw      ?                       ; Old stack segment

;---------- Task State Segment -----------------
                ALIGN 16

tsst1           TSS <>
io_map1         db      512 dup (0)             ; I/O Port 0000h ~ 0FFFh
                dw      0FFFFh                  ; Not protection

TSSLIMIT        equ     $-tsst1-1

                ALIGN 16
tsst2           TSS <>
io_map2         db      512 dup (0)             ; I/O Port 0000h ~ 0FFFh
                dw      0FFFFh                  ; Not protection


_DATA   ENDS

;---------- Stacke Segment ---------------------
; DGROUPに属し, DS or SS:SP(16bits) でアクセス

_STACK  SEGMENT PARA STACK USE16 'STACK'
                db      512 dup (?)     ; 特権LEVEL0用TASK2スタック
stack_top2      LABEL   WORD
                db      512 dup (?)     ; リアル,特権LEVEL0用スタック
stack_top       LABEL   WORD
_STACK  ENDS

;

;---------- 16 bits code segment ------------------------------------------
; リアルモード, またはプロテクトモード特権LEVEL0 16bitsコードで実行される
_TEXT   SEGMENT PARA PUBLIC USE16 'CODE'
        ASSUME  cs:_TEXT,ds:DGROUP

;---------- Goto Protect ----------------------
                                        ; リアルモードからプロテクトモードへ
                                        ; 移行SEL:32bits OFFをpushしてからjmp
GotoProtect     PROC FAR
;;      pushf
        push    eax
        cli                             ; 割り込み禁止
        mov     ax,DGROUP
        mov     ds,ax
        lgdt    [gdt_ptr]
;--- Protect Mode ---
        mov     eax,cr0
        or      al,1
        mov     cr0,eax                 ; プロテクトモードへ
        jmp     $+2                     ; プリフェッチクリア
        mov     ax,RDATA_SEL
        mov     ss,ax                   ; デフォルトセグメント設定
        mov     ds,ax
        mov     es,ax
        mov     ax,TSS1_SEL
        mov     [gdt_tss1].access,89h   ;非busyに設定
        ltr     ax                      ; TSS 設定
        pop     eax
;;      popf                            ; 割り込み許可
        db      66h                     ; 32bit code prifix
        ret                             ; far ret
GotoProtect     ENDP                    ; pushした アドレスへ


;---------- Goto Real -------------------------
                                        ; リアルモードからプロテクトモードへ
                                        ; 移行16bits OFFSETをpushしてからjmp
GotoReal        PROC NEAR
        pushf
        cli
        push    eax
        mov     ax,RDATA_SEL
        mov     ds,ax                   ; リアルモードでも許されるセグメント
        mov     es,ax                   ; をセット
        mov     fs,ax
        mov     gs,ax
        lgdt    [oldgdt_ptr]
        mov     eax,cr0
        and     al,0FEh
```

```
        mov     cr0,eax                 ; リアルモードへ
        jmp     $+2                     ; キャッシュオフ
        @Jmpfw  _TEXT,OFFSET _TEXT:greal_1
                                        ; csをセットするためfar jmp
greal_1:
        mov     ax,DGROUP
        mov     ss,ax                   ; es,fs,gsの値はRDATA_SELのまま
        mov     ds,ax
        pop     eax
        popf                            ; 割り込み許可
        ret                             ; near ret
GotoReal        ENDP


;---------- Privilege TEST ---------------------
start:
        mov     ax,DGROUP
        mov     ds,ax
        mov     [old_ss],ss
        mov     [old_sp],sp
        mov     ss,ax
        mov     esp,OFFSET DGROUP:stack_top
        call    Init386                 ; ディスクリプタ等の初期化

        push    DWORD PTR PCODE_SEL
        push    DWORD PTR (OFFSET ProtectMain - poffset_base)
        jmp     GotoProtect             ; プロテクトモード ProtectMainへ

;--- Return real mode DOS ---
pt_err:

        call    PutMsg
pt_exit:                                ; リアルモードで実行される
        mov     ss,[old_ss]
        mov     sp,[old_sp]
        sti
        mov     ax,04C00h               ; MS-DOSへ復帰
        int     21h

;--- Put Message ------------
; IN : ds:dx = message pointer
; RET: cx = print messagelength
PutMsg  PROC
        push    bx
        push    di
        push    es
        cld
        xor     al,al
        push    ds
        pop     es
        mov     di,dx
        mov     cx,0FFFFh
   repnz scasb
        not     cx
        dec     cx
        mov     bx,1
        mov     ah,40h
        int     21h
        pop     es
        pop     di
        pop     bx
        ret
PutMsg  ENDP


;---------- TSS2 Intialize
; 仮想86テストプログラムを動作させるためには
; このProceureを LIST 8と入れ替える必要があります

Tss2Init PROC NEAR
        mov     bx,OFFSET DGROUP:tsst2
        mov     [bx].TSS.tss_cs,PCODE_SEL
        mov     ax,RDATA_SEL
        mov     [bx].TSS.tss_ds,ax
        mov     [bx].TSS.tss_ss,ax
        mov     [bx].TSS.tss_fs,ax
        mov     [bx].TSS.tss_gs,ax
        mov     [bx].TSS.tss_es,ax
        mov     [bx].TSS.tss_eip,OFFSET task2
        mov     [bx].TSS.tss_esp,OFFSET DGROUP:stack_top2
        mov     [bx].TSS.tss_eax,'B'
        mov     [bx].TSS.tss_edi,160 * 10
        mov     [bx].TSS.tss_eflags,0202h
        mov     [bx].TSS.tss_cr3,0
        ret
Tss2Init ENDP

;---------- Intialize 386
                                        ; 386初期化
Init386 PROC NEAR
        sgdt    [oldgdt_ptr]            ; リアルモードにおける
                                        ; GDTRを保存
                                        ; ディスクリプタBASEを初期化
        mov     dx,_TEXT
        mov     ax,0
        mov     bx,OFFSET DGROUP:gdt_rcode
        call    SetDescBase
        mov     dx,DGROUP
        mov     bx,OFFSET DGROUP:gdt_rdata
        call    SetDescBase
        mov     dx,_PTEXT
        mov     bx,OFFSET DGROUP:gdt_pcode
        call    SetDescBase
        mov     dx,_PDATA
        mov     bx,OFFSET DGROUP:gdt_pdata
        call    SetDescBase
                                        ; GDTR値をセット
        mov     dx,DGROUP
        mov     ax,OFFSET DGROUP:gdt
        call    LinearAddr
        mov     [gdt_base],edx
                                        ; LDTR値をセット
        mov     dx,DGROUP
        mov     ax,OFFSET DGROUP:ldt1
        mov     bx,OFFSET DGROUP:gdt_ldt1
        call    SetDescBase
        mov     dx,DGROUP
        mov     ax,OFFSET DGROUP:ldt2
        mov     bx,OFFSET DGROUP:gdt_ldt2
        call    SetDescBase
                                        ; TSSディスクリプタの初期化
        mov     dx,DGROUP
```

```
        mov     ax,OFFSET DGROUP:tsst1
        mov     bx,OFFSET DGROUP:gdt_tss1
        call    SetDescBase
        mov     dx,DGROUP
        mov     ax,OFFSET DGROUP:tsst2
        mov     bx,OFFSET DGROUP:gdt_tss2
        call    SetDescBase
                                        ; TSSの初期化
        mov     bx,OFFSET DGROUP:tsst1
        mov     ax,RDATA_SEL
        mov     edx,OFFSET DGROUP:stack_top
        mov     [bx].TSS.p0ss,ax
        mov     [bx].TSS.p1ss,ax
        mov     [bx].TSS.p2ss,ax
        mov     [bx].TSS.p0esp,edx
        mov     [bx].TSS.p1esp,edx
        mov     [bx].TSS.p2esp,edx
        mov     [bx].TSS.tss_ldt,LDT1_SEL
        mov     bx,OFFSET DGROUP:tsst2
        mov     edx,OFFSET DGROUP:stack_top2
        mov     [bx].TSS.p0ss,ax
        mov     [bx].TSS.p1ss,ax
        mov     [bx].TSS.p2ss,ax
        mov     [bx].TSS.p0esp,edx
        mov     [bx].TSS.p1esp,edx
        mov     [bx].TSS.p2esp,edx
        mov     [bx].TSS.tss_ldt,LDT2_SEL
        call    Tss2Init
        ret
Init386 ENDP

LinearAddr      PROC NEAR
        movzx   eax,ax
        movzx   edx,dx
        shl     edx,4
        add     edx,eax
        ret
LinearAddr      ENDP

                                        ; ディスクリプタBASEをセット
SetDescBase     PROC NEAR
        call    LinearAddr
        mov     [bx].DESC.base_lo,dx
        ror     edx,16
        mov     [bx].DESC.base_mid,dl
        ret
SetDescBase     ENDP

_TEXT   ENDS

;---------- 32 bits Protect mode data segment ------------------------------

_PDATA  SEGMENT PARA PUBLIC USE32 'PDATA'
_PDATA  ENDS

;
;---------- 32 bits Protect mode code segment ------------------------------
; プロテクトモード特権LEVEL0 32bitsコードで実行される

_PTEXT  SEGMENT PARA PUBLIC USE32 'PCODE'
        ASSUME cs:_PTEXT,ds:_PDATA
poffset_base    LABEL   BYTE

;--- プロテクトモード・メイン処理
ProtectMain     PROC
        mov     ax,PDATA_SEL
        mov     ds,ax
        jmp     ttest                   ; testモジュールに制御が移る
RetReal::
        ;リアルモードへ移行して終了
        push    WORD PTR OFFSET pt_exit
        @Jmpfw  RCODE_SEL,OFFSET GotoReal
ProtectMain     ENDP

;--- eaxの内容を16進画面出力
HexPut  PROC NEAR                       ; eax(32bits) 16進 表示
        push    eax
        push    edx
        mov     dx,TVRAM_SEL            ; テキストVRAM 特権LEVEL0
        mov     es,dx
        mov     ecx,8
        mov     edx,eax
        cld
        .REPEAT
            rol edx,4
            mov eax,edx
            and eax,000Fh
            .IF (al >= 10)
                add eax,'A' - '0' - 10
            .ENDIF
                add eax,'0'
            stosw
        .UNTILCXZ
        add     edi,2
        pop     edx
        pop     eax
        ret
HexPut  ENDP

_PTEXT  ENDS
        END     start
```

・・・プログラムリスト2 TT386.INC・・・

```
;-----------------------------------------------------------
;       386 Task Switch TEST Program
;               Include file
;                               1993 10/01 ZOBplus Hayami
;-----------------------------------------------------------
_TEXT   SEGMENT PARA PUBLIC USE16 'CODE'
_TEXT   ENDS
_DATA   SEGMENT PARA PUBLIC USE16 'DATA'
_DATA   ENDS
_STACK  SEGMENT PARA STACK USE16 'STACK'
_STACK  ENDS
_PTEXT  SEGMENT PARA PUBLIC USE32 'PCODE'
_PTEXT  ENDS
_PDATA  SEGMENT PARA PUBLIC USE32 'PDATA'
_PDATA  ENDS
```

```
DGROUP  group   _DATA,_STACK

DESC            STRUC
limit           dw      0       ; Segment Limit low
base_lo         dw      0       ; Base (0 ~ 15)
base_mid        db      0       ; Base (16 ~ 23)
access          db      0       ; Access byte
atrlim          db      0       ; Attribute & Limit high
base_hi         db      0       ; Base (24 ~ 31)
DESC            ENDS

GDESC           STRUC
offset_lo       dw      0       ; Offset (0 ~ 15)
selector        dw      0       ; Selector
dcount          db      0       ; Dword Count
access          db      0       ; Access byte
offset_hi       dw      0       ; Offset (16 ~ 31)
GDESC           ENDS

TSS             STRUC
link            dw      0
                dw      0
p0esp           dd      0
p0ss            dw      0
                dw      0
p1esp           dd      0
p1ss            dw      0
                dw      0
p2esp           dd      0
p2ss            dw      0
                dw      0
tss_cr3         dd      0
tss_eip         dd      0
tss_eflags      dd      0
tss_eax         dd      0
tss_ecx         dd      0
tss_edx         dd      0
tss_ebx         dd      0
tss_esp         dd      0
tss_ebp         dd      0
tss_esi         dd      0
tss_edi         dd      0
tss_es          dw      0
                dw      0
tss_cs          dw      0
                dw      0
tss_ss          dw      0
                dw      0
tss_ds          dw      0
                dw      0
tss_fs          dw      0
                dw      0
tss_gs          dw      0
                dw      0
tss_ldt         dw      0
                dw      0
                dw      0
iomap_base      dw      0068h
TSS             ENDS


@Jmpf           MACRO seg,off
        IF @WordSize EQ 2
                db      066h
        ENDIF
                db      0eah
                dd      off
                dw      seg
                ENDM

@Jmpfw          MACRO seg,off
        IF @WordSize EQ 4
                db      066h
        ENDIF
                db      0eah
                dw      off
                dw      seg
                ENDM

@Callf          MACRO seg,off
        IF @WordSize EQ 2
                db      066h
        ENDIF
                db      09ah
                dd      off
                dw      seg
                ENDM

@Callfw         MACRO seg,off
        IF @WordSize EQ 4
                db      066h
        ENDIF
                db      09ah
                dw      off
                dw      seg
                ENDM


                EXTERNDEF       RCODE_SEL    : ABS
                EXTERNDEF       RDATA_SEL    : ABS
                EXTERNDEF       PCODE_SEL    : ABS
                EXTERNDEF       PDATA_SEL    : ABS
                EXTERNDEF       LDT1_SEL     : ABS
                EXTERNDEF       LDT2_SEL     : ABS
                EXTERNDEF       TSS1_SEL     : ABS
                EXTERNDEF       TSS2_SEL     : ABS
                EXTERNDEF       TVRAM1_SEL   : ABS
                EXTERNDEF       TVRAM2_SEL   : ABS
                EXTERNDEF       TASKGATE_SEL : ABS

                EXTERNDEF       gdt_tss1     : DESC
                EXTERNDEF       gdt_tss2     : DESC
                EXTERNDEF       stack_top    : WORD
                EXTERNDEF       stack_top2   : WORD
                EXTERNDEF       tsst2        : TSS

                EXTERNDEF       HexPut       : NEAR
                EXTERNDEF       RetReal      : NEAR
                EXTERNDEF       ttest        : NEAR
                EXTERNDEF       task2        : NEAR
```

連載

The Fantaastic VB World

第8回
～グラフィックス自由自在～

●著者 井上広大 Koudai INOUE

感情ではなく理性で

今回は，いつもとは違った方向から入ってみましょう．テーマは「感情ではなく理性を！」です．

最近，長引く不況のため，大声で環境保護キャンペーンする企業もめっきり減ったため，忘れられた感もありますが，現在最大の問題のひとつに環境問題があげられます．一説によれば，人類が150年以内に滅びる確率は90％以上だそうです．筆者は，このような時代に重要なものは「理性」，あるいはその力に裏打ちされた「想像力」ではないかと考えています．

コンピュータを使う上でも環境問題について貢献することもできます．

第一にペーパーレス．ネットワークとコンピュータが発展すれば，ほとんどの書類はペーパーレスになります．これで，パルプ製造のための森林伐採を減らすことができます．

第二に，人間の移動の減少によるエネルギー節約．在宅勤務やホームショッピングを可能にするコンピュータは，電気自動車よりもはるかに電力を節約するので，大きなエネルギー節約をもたらすでしょう．

そして第三に，効率的なエネルギー使用．いっさいの無駄が出ないようなAIによるエネルギー管理．すべての家庭にまで，このようなシステムがAIによるエネルギー管理を行うようにします．そうすれば，エネルギー状況は大幅に改善されるはずです．

しかし，コンピュータの世界は安直な性能競争，そして，ユーザのコピー問題など，レベルの低い問題が話題になるばかりで，エネルギー革命につながることなどは真面目に考えられてはおりません．

メーカは利益優先の製品開発をやめ，リサイクルや流用を可能にする製品を作ってください（結構，DOS/Vマシンはリサイクルできます）．ユーザも，安易なコピーでソフトメーカの足を引っ張るのではなく，自分こそがコンピュータ文化の次の担い手であるという自覚を持ちましょう．

コンピュータは本当に便利なものです．そして，さらに正統な進化をすることによって，大きな産業革命（消費革命）を起こすことができる可能性を持っているのです．

『とにかく「感情」なく，「理性」で考えるように！』が筆者からの提言です．

Visualこそコンピュータ普及の鍵

さて，話をVBに戻しましょう．

VBも消費革命に貢献する可能性のある言語です．といっても，いちばん消費を減らすのはエネルギーではなく，時間ですね．短時間でWindowsアプリケーションを作ることができるわけです（もちろん，そういう意味で電力消費も多少は減らせます）．また，少し前ま

でなら作成に時間のかかったGUIも，VBでは簡単に実現することができます．コンピュータをさらに普及させ，便利なツールとして認知してもらうためには，グラフィカルなユーザインターフェースは不可欠となります．こうした意味においても，GUIアプリケーションを作っていくことは重要です．

というわけで，今回はVBにおけるグラフィックの話ということになります(うまい展開だ！)．

VBのグラフィックコントロール

実は，VBのグラフィック機能は大したものではありません．GUIとはいっても，入力用のボックスを出したり，リストを出したりといった作業は，テキストを扱うものであるため，Visual処理も比較的簡単なのです．ところが，ビットマップを表示し，これを変化させたり，アニメーションさせたりといった機能は，VBも苦手です．Visualなどという名前がついているくせにグラフィック機能が弱いのは，問題があると思います．アメリカではすでに発表されている，次のバージョンにしても，この辺の事情は変わりません．そして，こうした問題は別売のコントロールなどでカバーされる傾向にあるようです．

しかし，基本的なグラフィック機能の使い方はマスターしておく必要があります．そして，これをポジティブに使っていくことによって，さらなる機能強化をメーカー側に訴えていくことが重要だとおもいます．

さて，それではVBに用意されているグラフィック関係のコントロールを復習しておきましょう．

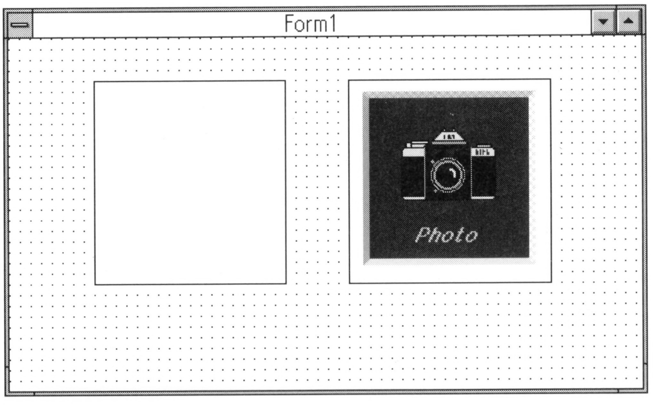
[グラフィック関連のコントロール説明]

1.ピクチャーボックスコントロール

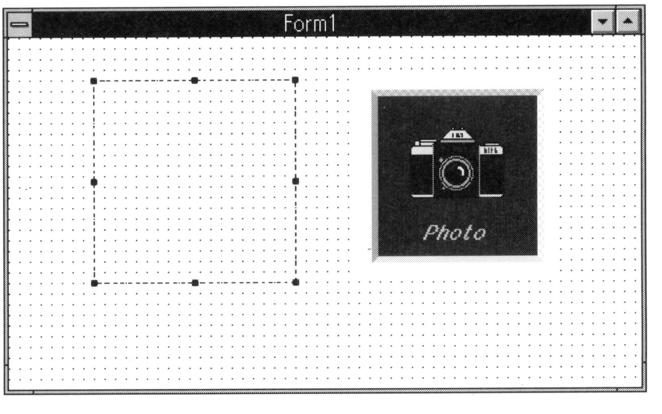
図1を見てください．ビットマップ，アイコン，メタファイルなどのグラフィックを表示するのが，ピクチャーボックスの役割です．

また，ピクチャーボックスは，MDIフォームのクライアント領域に配置できるフォームであることにも注目してください．ツールバーやステータスバーを作成する際に，このコントロールを使ってクライアント領域

▶図1●
ピクチャーボックスの例．
ピクチャーボックスには，グラフィック系のメソッドによるグラフィック表示の他，テキストを表示したり，オプションボタンをまとめるなどのコンテナの役割を与えることも可能だ．



▶図2●
イメージコントロールの例．
ピクチャーボックスによく似ているのは異論がないだろう．ピクチャーボックスよりも高速に処理を行えるので，アニメーションなどを行う場合には，イメージコントロールを使った方がよい．



の上下部分にあるコントロールをグループ化できます．

2.イメージコントロール

ピクチャーボックスに似ているので，間違われやすいのがイメージコントロールです．使い方も，重複する部分もあるので，気をつけてください(**図2**)．

とりあえず，イメージコントロールは，ピクチャーボックスの持つ機能を減らし，高速化を図ったものと考えておけばよいでしょう．

3.シェイプコントロール

図3を見れば一目瞭然です．方形や円をフォーム上に描くためにのみ使用されます．

4.ラインコントロール

これもシェイプコントロールと同系の命令です．LINE命令をあらかじめ実行するために利用されます(**図4**)．

アニメーションの作成入門

基本的な説明はこれぐらいにして，さっそく実際にプログラムを作ってみましょう．とりあえず作成するのは，アニメーションです．

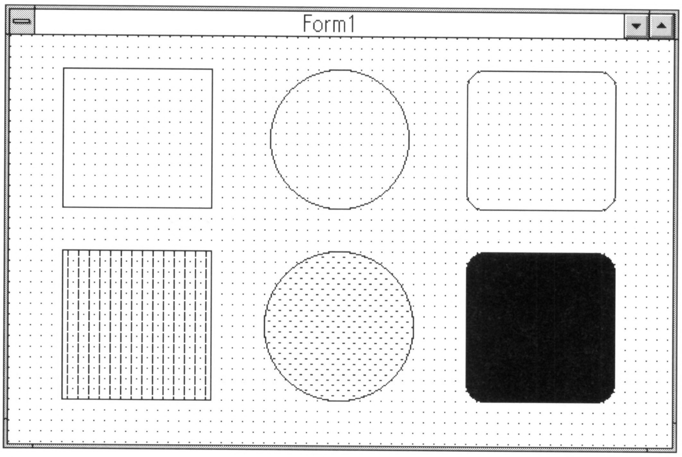
アニメーションを作るためには，イメージコントロールと，これまでの連載ではあまり登場しなかったタイマコントロールを利用することになります．

[アニメーション描画プログラムの作成手順]

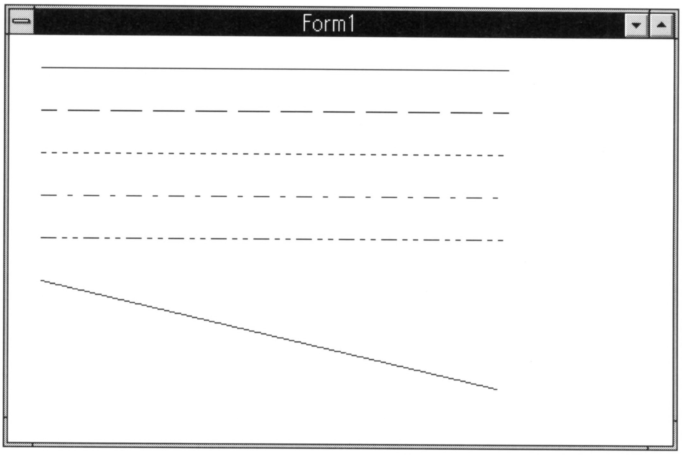
1.アニメーション素材を用意する

絵の得意な方は，自分でビットマップ(またはアイコン)を作ってください．ここでは，あらかじめVBに用意されている豊富なアイコンを利用することにします(**図5**)．

▶ 図3●
シェイプコントロールの例．
昔からベーシックにあった，CIRCLE命令などを，フォーム描画と同時に描くもの．当然ながら，コードにも同じ機能を持つメソッドが用意されている．目で確認しながら描画できるのが特徴だ．



▶ 図4●
ラインコントロールの例．
ほとんど説明の必要はないだろう．最も基本的なグラフィックメソッド．



2.コントロールの配置

通常のコントロール配置と同じように，イメージコントロールをフォーム上に置きます(**図6**).

その他に，必要となるボタン(この例の場合“アニメーション開始”，“アニメーション終了”，さらにタイマコントロール)も用意しておきます.

3.コーディング

通常，アニメーションを行う場合，MOVEメソッドを使います．これで，イメージコントロールの位置をずらしていくのです.

MOVEメソッドを使う場合には，次の点に注意してください.

〈MOVEメソッド使用上の注意〉

・Twip単位の座標系

MOVEメソッドでは，図形の位置や長さを表す単位に，Twipという単位を使用する．1Twipは，フォントで使われている1ポイントという単位の1/20の長さで，たとえば1センチは，567Twipということになる．画面上での長さは，画面のサイズや解像度によって異なる.

・基準

画面上でフォームを動かす場合は，画面の左上が基準(0，0)となる.

フォーム内でイメージコントロール等を動かす場合は，フォームの左上が基準(0，0)となる.

・絶対移動

絶対座標を指定して，その場所にコントロールを動かす場合は，次のようにする.

(例)

```
Field1.Move 50, 100
```

これは，Field1というコントロールを(50,100)の座標に表示することを意味する.

・相対移動

現在コントロールがある場所からの相対位置座標を利用して移動を行う場合は，次のようにする.

(例)

```
Fd1.Move   Fd1.Left + 50, Fd1.Top + 100
```

これは，Fd1という名前のコントロールを，右に50Twip，下に100Twipだけ動かすことを意味する(Left+50は左からの位置を50だけプラスするということ).

次に，このMoveメソッドを使ってのコーディング例を見てください(**図7**，**図8**).

単純なアニメーション表示だけであれば，このような処理で十分である.

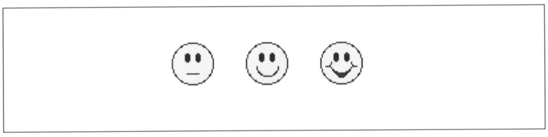
4.アニメーションの完成

あとは，アニメーション描画を終らせるコマンドボタンをコーディングすれば完成です.

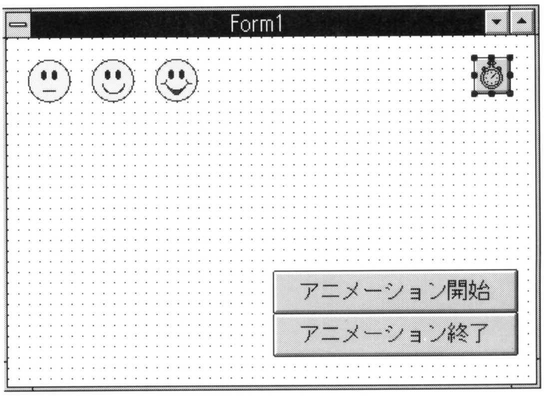
アニメーションの枚数を増やすなどの工夫を行えば，さらになめらかなアニメーションを楽しむことができますし，アニメーション用のグラフィックをロードできるように改良すれば，簡単なアニメーションチェックプログラムとしての実用も可能となります.

その他のグラフィックメソッド

アニメーション関連ということで，Moveメソッドを



▲図5●
アイコンライブラリには，アニメーションとして使えそうなもの(同じものが変型しているもの)がいくつかある．Windowsなどで使われる，ほとんどのグラフィックボタンなどは用意されているので，便利だ.



▲図6●
イメージコントロールをアニメーションで使う場合，初期の配置位置には大きな意味がない場合もある．グラフィック系のメッソドを使うことによって，アニメーションの表示位置を変化させる場合があるからだ.

中心に説明いたしましたが，この他にもグラフィック系のメソッドはいろいろあります．Moveメソッドよりもわかりやすいものばかりなので，簡単に説明しておきます．

［グラフィック系のメソッド］

1.Cls

すべてのグラフィックとPrintによる出力をクリアするメソッド．

2.PSet

ピクセル単位で，色の設定を行うメソッド．

3.Point

指定したピクセルの色の値を返すメソッド．

4.Line

線描画，ボックス，ボックスフィルを行うメソッド．

5.Circle

円，楕円，円弧を描画するメソッド．

グラフィック系のメソッドを使うと，グラフィカルコントロールよりもメモリを節約することができます．アプリケーションを起動してからでないと，動作が確認できないという欠点はありますが，これまでのベーシックユーザの方であれば，何の問題もなく使えるのが，これらのグラフィック系メソッドです．

しかし，グラフィック用のメソッドが少ないというのは，Visualを売り物にしている割には情けないですね．プログラミングに十分な時間を費やせない方は，別売のグラフィカルコントロール集などを利用した方がよいでしょう．

ゲームの作成

さて，グラフィックの使い方を説明してきました．一見面倒くさそうですが，よくよく見てみると，非常に単純です．Moveメソッドを覚えるだけで，いろいろできてしまうからです．そこで，突然ですが，ゲームを作ってみることにしましょう．ゲームは，よくある15ゲームの変型版です．

詳しい説明は次号に回すとして，とりあえずフォームデザインを見てください(**図10**)．

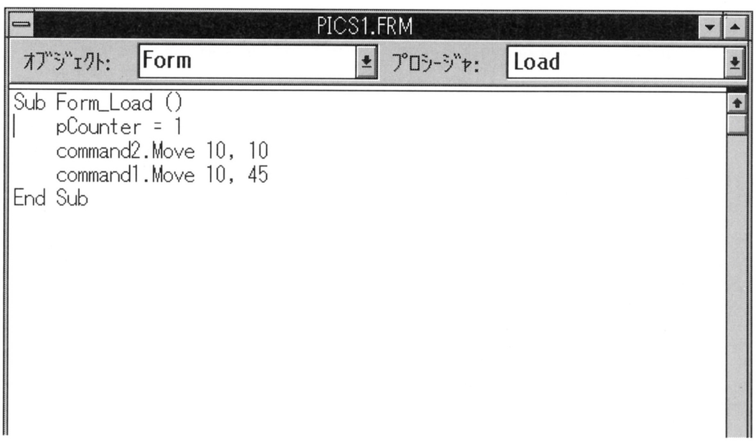
ゲームなんか，とバカにする方もいらっしゃるでしょうが，筆者はグラフィック制御のテクニックを身につけるには，簡単なゲームを作ってみるのがいちばんだと考えております．ぜひ，皆さんも面白い題材を見つけて，ゲームを作ってみてください．

次回の予告

次号は，今回最後でデザインした，15ゲーム変型版のコーディング等を行いながら，VBのグラフィック制御応用編を説明いたします．

題して「ゲームプログラムは身を助ける！」です．

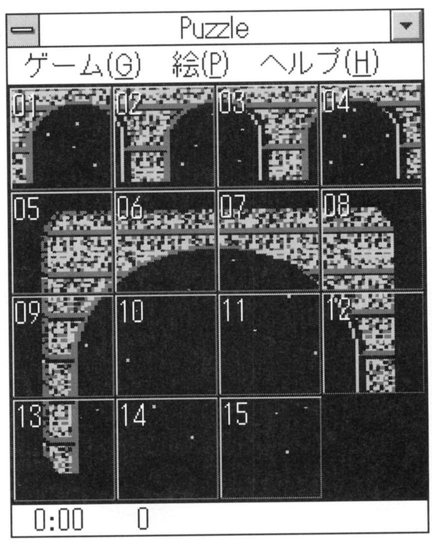
▲図7●
“アニメーション開始”がクリックされたときに実行されるプログラム．
Moveメソッドの絶対移動を用いて，アニメーションを適当な場所に置く．



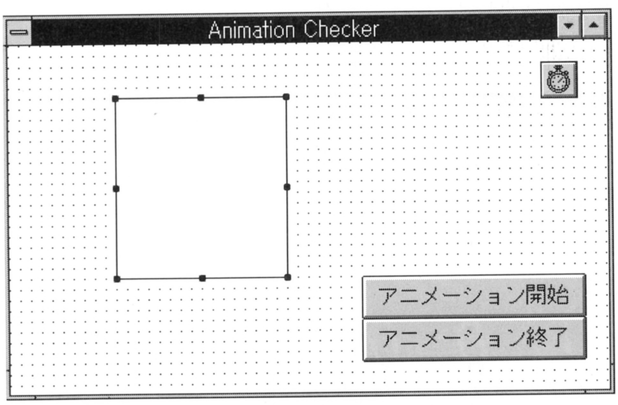
```
PICS1.FRM
オブジェクト: Timer1    プロシージャ: Timer

Sub Timer1_Timer ()
    'Static pCounter As Integer
    If Mair.Left > Form1.ScaleWidth Then Mair.Move 0
    If Mair.Top < 0 Then Mair.Move Mair.Left, Form1.ScaleHeight
    '顔のピクチャー移動
    Mair.Move Mair.Left + 10, Mair.Top + 3
    Select Case pCounter
      Case 1
          '顔1のピクチャーを表示
          Mair.Picture = Face1.Picture
      Case 2
          '顔2のピクチャーを表示
          Mair.Picture = Face2.Picture
      Case 3
          '顔3のピクチャーを表示
          Mair.Picture = Face3.Picture
    End Select
    'カウンター処理
    pCounter = pCounter + 1
    If pCounter > 3 Then
      pCounter = 1
    End If
    'Print pCounter
End Sub
```

▲図8●
このプログラムは，タイマコントロールに記述してある．タイマコントロールによって，時間をカウントし，それによってアニメーションを動かすという形式にしてあるからだ．



▲図10●
15ゲームの変型版デザインの図．
基本的な遊び方は，15ゲームと同じで，タイルをスライドさせて，すべてを元の位置に戻せばよいというもの．グラフィック関係のさまざまな技法を使っている．



▲図9●
先のプログラムに手を入れて，簡易アニメーションチェッカにした例．このような改良は，数分でできるのが，VBの底力である．

大きなお店と小さなお店

ある商店街からちょっと離れたところに，大きなスーパーマーケットができるという噂がたちました．もし，そこにスーパーマーケットができると，商店街は大きなダメージを受けるのは必至です．場合によっては，閉店に追い込まれる店もあるかもしれません．これまで，和気あいあいとまでいかなくても，安穏と商売をやってきただけに，これは大きな衝撃でした．

噂は現実となり，ある店主は，共同でスーパーマーケットができることを反対しようと意見を唱えました．頑固な店主は，意地でも作らせないとスーパーマーケットの建設予定地に座り込みをはじめました．しかし，そんな抵抗も虚しく，スーパーマーケットが開店の日を迎えました．．

共同戦線も頑固な座り込みもまったく効果がなかったわけです．

水が低い方に流れるように，消費者は，同一の商品であれば安い方を選びます．また，スーパーマーケットはさまざまな（業種の）商品を陳列することができる強みがあります．とりあえず行けば，なんでも揃うという利点もあるし，社員教育は万全です．閉店間際には，投げ売りまでやってみたり，折り込み広告を入れて激安価格のイベントを繰り広げるのです．

こうして一カ月の歳月が流れました．商店街は，店主達が予測した通りお客様の大半をスーパーマーケットにとられてしまい，閉店を余儀なくされた店舗も目立ちはじめました…．

魚屋さん

ところが，商店街のなかほどにある魚屋さんだけは違っていました．一時は売り上げが極端に減ったものの，今では商店街全盛期を越えるほどの繁盛ぶりを見せています．

この魚屋さんは，**独自のサービス**を展開し始めたのでした．元々，この魚屋さんは，ときどき突飛なサービスをすることが有名で，少なからずファンがいました．このファン層に支えられてなんとか生きてこられましたが，やはりスーパーにはかないません．ならば，ウチならではのサービスを展開しようと考えたのです．**小さいならではのゲリラ戦法**です．

冬なら鍋．鍋ならタラをつかったチリ鍋，牡蠣をつかった土手鍋．最近の奥様方の案外不精なところに目をつけ，それらの鍋の魚に野菜をつけて販売を開始しました．しかし，ここまでは，スーパーでもやっています．しかし，魚屋の店主は，**自分のところにしかできないアイディアをひねり出しました**．

普通，タラチリに使うタラは上がった（死んでしまった）タラを用います．タラという魚は海底を這う魚だけに，体温が高くないと生きていけません．したがって，体温をある程度高めに維持しているため，釣ってすぐに冷やして保存してたとしても，体温が高いためにすぐに内蔵が腐敗してしまうのです．

こういったタラは，キロ単価が安いため，惣菜と抱き合わせて鍋セットなどを作ると，かなりの利幅が見込めます．この辺をよく研究してか，スーパーではかなり以前から，こうしたセット販売をやっていました．

ところが，魚屋さんが考えだしたのは，そういった上がりもののタラを使うのではなく，**活け**（“いけ”と読む）のタラを使うことでした．活けのタラは，上がりのタラよりももちろん高価です．しかし，味は絶品で，湯通ししても身はプリプリしていて，フグを食べているような錯覚さえ覚えます．

それに加え，魚屋さんは伊達に永年**魚河岸通い**をしていたわけではありません．馴染みの店なら安くしてくれます．また，本当に旨いタラチリを作るならば，タラの眼球周辺も出汁として使う．このあたりも研究し，盛り付けも有名な料理屋などを食べ歩き，研究しました．こうして，魚屋さんは**独自のカラー**，つまり本当に美味しい鍋の素材と，見栄えのする盛り付けを提供することにより，確実に客層を掴んでいきました．ただし，最初は血へ

CHAPTER4

大きなお店と小さなお店

O2同好会

オーツー

ドが出るくらいの努力をしたのは明らかです．もちろん，原価すれすれの超安値で対抗しなくてはならなかったのですから．

かなわないか？

巨大な資本力を活かし，マニュアルなどの大量印刷による原材料費のコストダウン．目を見張るプレゼンテーション効果抜群の広告．同じ内容なら，どうしたって小規模店舗がかなうわけがありません．しかし，自分で努力もせずに他人のせいにして尻尾を巻くヤツは嫌いです．

財力と人海戦術でくるなら，こちらはゲリラ戦法でうちならではのカラーで戦いたい．

そして，いつかは巨大な資本を投下できる企業に成長し，会社のコンセプトでもある

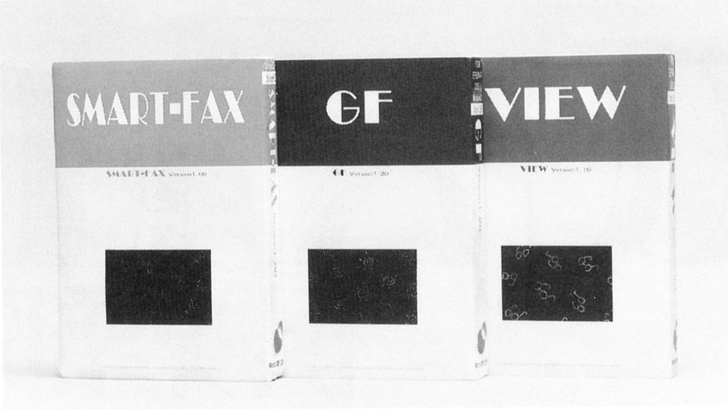
酸素のように「あって当たり前」

と呼ばれるようないソフトウェアを作って，画期的な価格で販売したい．これが私の夢です．

今回，起死回生の一撃をねらって，GF/Winを開発，しかも従来のGF98に加え新開発したGF/Vを丸ごと1パッケージにし，なおかつ全開パワーアップセールと銘打って，**破格の6,800円**で店頭にならべました．

これこそ，血ヘドが出そうでした．私も含め，従業員一同は徹夜につぐ徹夜で，なんとか期日に間に合わせようと必死で戦いました．結果，社員一丸となった成果があらわれ，奇跡ともいえるスケジュールを見事にこなしました．しかし，GF/Winのみを除いて….

GF/Winの開発は，6月から取り掛かっていました．スケジュール的には比較的余裕のあるもので，βテストに，約1カ月割り振れるようになっていました．しかし．実際に作業が進むにつれ，プログラマは，見込みが甘かったことに気がつきはじめたようです．プログラマは，それをひた隠し，Log Diskができるだけの*.EXEだけは見せてくれたものの，それから遅々として先に進みません．業を煮やした私は，彼にソースを要求しました．快く見せ



てもらったものの，内容は惨憺たるもので，これでは到底間に合うわけがないことが分かりました．

ともかく，建設的な事後処理を行うべく，即座に，**GF/Winの遅延アナウンス**をするとともに，**「ご登録ユーザー様にはGF/Winは11月末に無料でご発送」**といった旨のシール作成にとりかかりました．しかし，チェックが甘く，出荷時に，大量のGFに**先程のシールが貼られていない**ことが判明．私は失神しそうになりました．

発送してしまった分に関しては，回収作業に取り掛かり，できるかぎりシールを自分で貼りにまわりました．それでも，この傷を癒すには相当の時間がかかることだろうと予測されます．

今まで，GFを丹念に育て，そして，ここにきて巨大なオーダーをいただきました．つまり，独自のカラーを出しきって戦い，そして勝利を納めつつあったのです．本当にうれしかった．しかし，シールが貼られていないGFをお買い求めになったお客様は激怒し，苦情の電話をとるたびに，**今まで築いてきた信用が足元からはガラガラと音を立てて崩れていく**ような気持ちがしました．お客様がお怒りなるのはごもっともで，パッケージには「GF/Win同梱包」などと書かれてありながら，パッケージにはGF/Winはないのですから．

転んだら

私は，**このままにしておくつもりはありません．**

シールを貼ったからいいだろうなどとも考えておりません．

この失敗をまっすぐに受け止め，事後処理をいたします．

開発計画をいいかげんにたてさせ，広告を先行させた責任はすべて私にあります．

責任のとりかたにあります．

迷惑をかけたから潔く会社をたたんで，とっととこの世からオーツーが消えてしまう．これは卑怯極まりなく，単なる逃げでしかありません．

私は，今回の失敗を糧にし，**二度と同じ失敗を繰り返さないこと**が第一のユーザサービスと心得ます．また，新規に購入されたお客様およびバージョンアップをしてくださったお客様皆様に，**無償にて新しいマニュアルを発送**いたしますことを，ここにお約束いたします．

オーツーは，まだまだ町の商店と変わらない規模の会社です．なかなか理想のようにはいきませんが，努力とアイディア，そしてウチならではのカラーを貫き，酸素たるソフトウェアをこれからも開発していきます．

どうか皆様．

言えた義理ではありませんが，まだまだヨチヨチ歩きの会社で前代未聞の大失敗をしてしまいましたが，どうか応援してくださいますよう，切にお願い申し上げます．

（日高 豊）

㈱オーツー
☎048-446-0290 / FAX 048-446-0291

Welcome to our UXⅡ倶楽部

パーソナルUNIXシステム「PANIX for AT Ver3.0」新発売

お客様からの熱いご要望にお応えし，SVR4の本格カーネルをもつパーソナルUNIXシステム「PANIX for AT Ver3.0」(PC/AT互換機用)を発売します．また，同時に「PANIX for 98」(NEC PC-9800シリーズおよびEPSON PC-386/486シリーズ用)をVer3.0にバージョンアップします．発売開始は'93年12月20日．標準価格は各4万8千円です．

また関連商品として，LAN接続用の「TCP/IPキット Ver3.0」(価格1万8千円)のほか，統合・充実が図られた「フリーソフトバイナリパックⅢ」「フリーソフトソースパックⅢ」(価格各1万円)を発売します．

販売方法はエー・アイ・ソフト㈱からの直接販売のみ．同時に現ユーザに対してのバージョンアップを現在受付中．技術的なサポートはBBS(NIFTY Serve, 日経MIX, PC-VAN)かFAXで受け付けます．

■PANIX for AT/for 98Ver3.0の特徴

低価格の本物(SVR4)

カーネルには業界標準のSystem/V 386 Release4.0の本格カーネルを採用し，メッセージを日本語化することで分かりやすい操作を実現．

開発環境(CC & GCC，ライブラリ)

標準で二つのコンパラ(標準CCと最新GCC 2.4.5)とライブラリ群を装備し，ソフトの開発・移植，XやLAN対応のプログラムが作成可能．

最新のGUI

(X11R5, 最新アクセラレータボード対応)

日本語化されたX11R5を採用し，最新のS3搭載のアクセラレータボードに対応することで，ワークステーション並みの速度を実現．

オンラインマニュアル(英語，日本語)

英語のフルオンラインマニュアルのほかに，主要コマンドは日本語マニュアルを添付．

インストール(自動 &マニュアル)

メッセージにしたがってフロッピーを入れ替えるだけの自動インストールと，細かな設定ができるマニュアルインストールが選択可能．

環境設定ソフト(Sysadm)

新規ユーザーの登録，パスワード変更，ファイルシステム管理などが，メニュー選択だけで行える日本語環境設定ソフト「Sysadm」を標準装備．

DOSとの共存(HD，FD)

フロッピーディスク，ハードディスクにあるDOSファイルを直接読み書き可能．

強靭なシステム(自動復旧)

突然の電源断，リセットが起こっても，システムの「自動復旧機能」によりユーザの負担が軽減．ファイルシステムをチェックし，自動修復後に正常立ち上げ．

■TCP/IPキット Ver3.0

PANIXからイーサネットを通じて他のPANIX，あるいはWSと接続するためのプログラム．Intel EtherExpress, NOVELL NE2000，3Com Etherlink Ⅱ(以上，AT用)，マクニカNE-2098，アライドテレシス SIC-98, SIC-98E, SIC-98ET, SIC-98NOTE-T, SIC-98NOTE-E, メルコEGY-98-2, EGY-98-Note(以上，98用)などに対応済．なお，ネットワークの構築にはシステム管理者レベルの知識が必要です．

■フリーソフトバイナリパックⅢ

実行可能なバイナリプログラム集．Gnu Nemacs 18.55-3.3.2，Nxeditor 2.02，Wnn 4.10，かんな 1.2やtcsh 6.02.00，gdb 4.10，lha 1.1などのツール類，アスキー日本語T_EX2.99-j1.7e，Ghostscript 2.41jなど17種類のフリーソフトを収めました．

■フリーソフトソースパックⅢ

フリーソフトバイナリパックⅢの全ソースと下記のXサーバおよびGNU関係のソースプログラム集

・X11R5のXサーバ(98用)
・Gnu C Compiler 2.4.5
・Gnu libg++ 2.4
・Gnu troff 1.06

■販売方法

製品の販売は直接販売のみとし，発送は'93年12月20日から行います．購入希望者は，郵便振替にて代金(税込み)を当社に振り込んでください．振替口座番号等は，以下の通りです．

郵便振替口座番号：「長野1 61275」
加入社名：「エー・アイ・ソフト株式会社」
金額欄：購入する商品の合計金額(税込)

PANIX for AT Ver3.0	4万9,440円
PANIX for 98 Ver3.0	4万9,440円
TCP/IPキット Ver3.0	1万8,540円
フリーソフトバイナリパックⅢ	1万0,300円
フリーソフトソースパックⅢ	1万0,300円

通信欄：購入する商品名とメディアの種別(AT, 98/3, 98/5)を記入

AT	AT対応(3.5インチ版のみ)
98/3	98対応の3.5インチ版
98/5	98対応の5インチ版

■バージョンアップ

・また，バージョンアップの発送は，'93年12月20日から行います．バージョン

アップ希望者は，郵便振替にて代金(税込み)を同社に振り込んでください．バージョンアップ期間は'94年1月31日まで．なお,「PANIX Ver1.0/2.0」「TCP/IP体験キット Ver1.0」からバージョンアップできるのは，AT版，98版のいずれか一方のみ．

振替口座番号などは，製品購入と同様．金額欄には，バージョンアップの合計金額(税込)，通信欄には，購入する商品名とメディアの種別を記入してください．

バージョンアップ料金は下記の通り．

PANIX Ver2.0から
- PANIX for AT Ver3.0　3万8,110円
- PANIX for 98 Ver3.0　1万0,300円

PANIX Ver1.0から
- PANIX for AT Ver3.0　3万8,110円
- PANIX for 98 Ver3.0　1万5,450円

TCP/IP Ver1.0から
- Ver3.0(AT用)　1万0,300円
- Ver3.0(98用)　5,150円

製品申し込み先

〒182 東京都調布市布田1-29-2
ビルディング川口6F
エー・アイ・ソフト㈱東京営業所PANIX係
☎0424-85-7417, FAX 0424-85-7554

〒390 長野県松本市中央2-1-27
本町第一生命ビル7F
エー・アイ・ソフト㈱ PANIX係
☎0263-36-1424, FAX 0263-36-5751

■お急ぎの方へ

郵便振替による方法は代金を振り込んでからお手元に商品が届くのに10日前後かかります．お急ぎの方は，郵便振替の控えのコピーをエー・アイ・ソフト㈱ PANIX係(松本市：住所は前述)まで郵送またはFAXしていただければ製品を発送いたします．

■技術サポート

UNIX自体に関するサポートはエー・アイ・ソフト㈱ PANIX係(松本市：住所は前述)もしくは下記BBS上で受け付けます(無料)．電話での対応は原則として行ないません．

NIFTY-Serve	FAISAP
日経MIX	aisoft/panix
PC-VAN	AISOFT

「福袋'94」をユーザ直販

WXⅡ⁺Ver.2.56(98版，J-3100版，DOS/V版)とMacintosh用日本語インプットメソッドWXⅡ for Macにアドインする製品「福袋'94」を，'93年12月20日より，上記製品の登録ユーザ様に対し直販いたします．価格は4千円で，登録ユーザ以外には販売を行わない予定です．尚，福袋'93ユーザには2千円でバージョンアップいたします．販売予定本数は約5万本です．

「福袋」は，同社が毎年末にWXIIシリーズのユーザに向けて販売している商品で，今年で3回目を迎えます．変換に必要な最小限の機能のみを搭載したドライバや，各種の新辞書，追加登録単語テキストを搭載する商品です．今年は新しくMacintosh用もラインナップに加わり，よりマルチプラットフォームを意識した商品構成になっています．

「福袋'94」は'93年11月20日より対象WXⅡシリーズの全登録ユーザにDM送付を開始し，'93年12月20より出荷を開始いたします．

■「福袋 '94」の内容

「福袋'94」for DOS /「福袋'94」for Mac

郵便番号テキスト
郵便番号から住所への変換を約2万8千語収録(ぽすたるガイド最新版準拠)

人名テキスト
日本人の姓名約8,300語収録(昨年度テキスト使用)

企業名テキスト
主な企業名約2千社を収録(福袋'92テキスト更新)

地名テキスト
日本の主な地名約5,300語を収録(福袋'92テキスト使用)

WXⅡミニシステム
WXⅡ⁺Ver.2.56より，変換に直接関係の無い機能をすべて削除することで，常駐サイズを100Kバイト未満，EMS使用時20Kバイト未満に低減しています(DOS版のみ)．
(昨年度同様Ver.2.56カーネル相当)

100K辞書
日常使用に不都合にならない必要最小限の単語を収録した極小辞書．

空辞書
全くゼロから辞書を育てたいユーザのための辞書．

医学用語辞書
NIFTY-Serveのエー・アイ・ソフトフォーラムにあるフリーソフトウェアの医学用語辞書(池田勇氏著作権所有)．

WX-WORD for Windows

12月3日より，日本人による日本人のための日本語Windowsワープロソフト「WX-WORD for Windows」が好評発売中です．定価は1万9,800円(税別)です．また,「WX-WORD foe Windows」の発売を記念して，初回限定1万本を特別価格1万4,800円(税別)で販売するキャンペーンを実施中です．

WXⅡ-Win Ver.2.1 バージョンアップ

日本語MS-Windows専用FEP WXⅡ-WinをVer.2.0からVer.2.1にバージョンアップし，'93年12月3日より発売中です．既に購入されているWXⅡ-Winユーザに対しては，「WX-WORD for Windows」の発売を記念して無償バージョンアップを行っております．ユーザ登録されていない方は，お早めにユーザ登録をお願いいたします．

WXⅡ⁺上級者御用達ツール

WXII+で使える面白いツール類を多数集めた「WXII+上級者御用達ツール」が，皆様のご要望に答えてパッケージ販売され，好評発売中．標準価格は，9,800円(税別)です．

エー・アイ・ソフト㈱　小山泰男

エー・アイ・ソフト㈱営業部
〒182　東京都調布市布田1-29-2
ビルディング川口6F
TEL　0424-85-7444
(ユーザサポート専用)
受付時間　10:00 ～ 12:00
13:00 ～ 17:00
月曜日～金曜日(祝祭日は除く)
FAX　0424-85-7554

❽（最終回）
しめくくりに

バックス通信
VACS REPORT 最終回

難しいお話

本誌に掲載されていた，中村正三郎氏の「電脳曼陀羅」連載中止に関して，パソコン通信などでいろいろ賑わっているようですが，深読みの私としては，変なことを思っています．それは，やっぱり日本では，世界に通用する本格的で，優秀なソフトウェアプログラマ（デザイナ）が出てこないな，ということです．この業界にも，どこかに“見えざる手”があり，その手はいつも特定のところに利益が集まるように仕組まれているなと，最近思うようになりました．

例えば，スーザン・ラマース/マイクロソフトプレス編『実録！ 天才プログラマー』（アスキー出版局）を読んでも，歴然としています．作者の地域的な問題や，ソフトウェア産業の歴史が浅いということも手伝ってか，それとも，日本人特有の“奥床しさ”も，あるのでしょうが．

強いていえば，かな漢字ソフトしかないのでしょうか．あの，TRONも現実的に世の中に出てくる前にスーパー301条という奴に，なかば潰された感があります．

いま，流行っているほとんどのアプリケーションは，欧米からのものです．また，低価格戦争を繰り広げ，その中で勝ち残ったものだけが，21世紀に“官軍”として名乗りをあげることができます．ほとんど体力勝負になってきました．

たしかに，よいアプリケーションの条件として低価格があるのでしょうが，それがすべてでないのも事実だと思います．よいものはよいと，価格の面のおいても，正当に評価されるべきものでしょう．と，いっても現実問題として，火中にいる者として，ある面弱気をいってもしようがないので，頑張るしかありません．

いままでは一種の，鎖国時代でした．それがある日，黒船がやってきて，日本では，そんなに無茶なことをやらないだろうと，高を括っていたところに，この現状です．

戦後，世界競争力のあった日本の輸出品が，次第にその競争力を失ってきて，その反動がいまを象徴している状態なのでしょう．

日本から，世界に通用するような，アプリケーションは，いつになったら出てくるのでしょうか．出なくてもいいのかな．

日本でWindows 3.1の発表があったときに，来日したMicrosoft社の最高責任者ビル・ゲイツ氏がNTTに，米国副大統領のゴア氏ほかが提唱している，エレクトロニクスーパーハイウェイ（情報のスーパーハイウェイ構想）を，日本でも一緒にやりましょうと，言ったとか，言わないとか．何でも，非常に高速な（中身は光通信とか）通信網を構築して瞬時にデータを転送するそうです．例えば，日本とアメリカでテレビ会議を行っても，リアルタイムであるために，今のテレビ電話の静止画程度のものではなく，動画を違和感なく双方向にデータが送れるものだそうです．そのときに提出された書類も，瞬時に相手側にドキュメントとなって送られるとか．そんな環境が，今のPC程度で構築できるとしたら，どんな世界になるのか見当がつきません．スケールが大きいな．

さらに，話題のバーチャルリアリティとかマルチメディアがあります．マ

ルチメディアとは何かと問うても，明確な答えは返ってこないでしょう．しかし，エレクトロニクスーパーハイウェイを最初に提唱したと言っている，ナム・ジュン・パイク氏は，マルチメディアとは「人間の五感を刺激するものである」と言っています．

ディスプレイをキャンバス代わりにし，音声をデジタルで録音して加工する．皮膚感覚は，バーチャルリアリティで再現する．まったく，その通りなのでしょう．

技術がある，ないの問題ではなく，そこにコンセプトがあるかないかの問題なのでしょう．最初に“思い”があり，それから技術がついていくのでしょう．これまでのイノベーションも，たぶんそのような姿で成り立ってきたのでしょう．

ソフトウェア業界に故本田宗一郎氏がいたらなと，つくづく思います．たぶん同等の人物はいるのでしょうが，世界の誰々といわれる人物はまだ出ていないのです．

いまの，政治ではないのですが，ソフトウェア業界の再編成が来たときに泣きをみないためにも，武装しておく必要があります．

簡単なお話

いま，OSやCPUも戦国時代です．と，いうより，いかに自分のところのソリューションを売り込むかに懸命です．PC/ATの世界に，Microsoft社は，Windows NTや，コードネームでChicago（シカゴ），Cairo（カイロ），SunSoft社は，SolarisのIntel版を，NeXT社は，NEXTSTEPのIntel版などを一斉に発売しています．PC UNIXの世界は乱舞しています．

CPUの世界では，IntelのPentium，IBMとMotorolaのPowerPC．これらはRISCチップのテクノロジーが組み込まれているとか．さらに，x86の互換CPUにはCyrix，AMD社のCPUなどがあります．

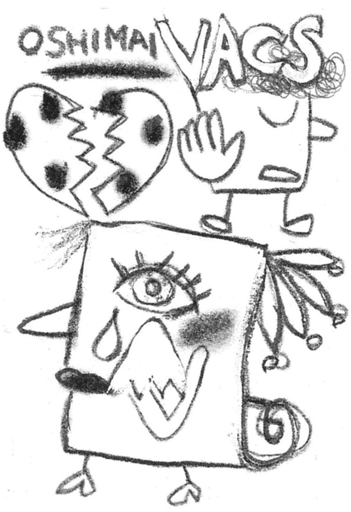
このようにPCの世界に，なだれ込んできた理由のひとつに，PCがひと頃のワークステーションと同等のスペックを持たなければならなくなってきたからです．

最近はあまり聞かなくなりましたが，「ダウンサイジング」という名のもとに，各メーカーがいかにPCを売り込むかに，凌ぎを削った時期がありました．

汎用機からWSに，そしてPCへとハードウェアの推移があり，それを追いかけるかのように，（OS）ソフトが追従する現象となりました．その結果として，現在UNIXベースで稼働している（重い）OSがPCへと，流入してきたのです．

しかし，現在PC上で最高速といわれているi486DX2/66MHzマシン（Pentumはまだ一般的ではないので）で快適に使用するためには少し無理があるようです．そのため，もっとパフォーマンスのよいCPUが必要になってきました．

そこで登場するのがCPU戦争です．我こそは，コストパフォーマンスに優れたCPUであると，どこのメーカーもいっていますが，最後にはマーケティング力がモノをいい，21世紀にもマーケティング力のあるところが官軍になります．



しかし，冷静に考えてみますと，現在でもハードウェア，OSに振り回されているがソフトメーカーです．新しいハードやOSが出てくるたびに，インストールやセッティングを行わなければなりません．そのつど，新しいOSを勉強しなければなりません．本来ならばOSメーカーが答えるべきことを，アプリケーションメーカーが答えているということも，いまだに多々あります．

それは，逆に考えればPCユーザの裾野が広がったとみるべきなのかもしれません．

こちらの話は，チップ1個いくらという世界なので，話としては分かりやすいかもしれません．

それでは

この「バックス通信」も，今回が最終回となりました．8回という短い間でしたが，ご愛読いただき，ありがとうございます．

ある日突然，このコーナーの前任者・新井から「お前，バックス通信を書け」と言われ，その夜（お酒の）勢いで引き受けたのが，そもそもの間違いであったのでした．私自身，文章を書くのは（得意というわけではないのですが），嫌いではなかったので引き受けたのですが，毎回テーマを決めることと，締め切りとの戦いでした（大袈裟かな）．

もしも，またお目にかかることがありましたらもう少し，面白い内容で書いてみたいと思います．

（バックス　友田　春芳）

●

㈱バックス サポートセンター
TEL　0427-24-9200
FAX　0427-28-6864

また，バックスでは，次のパソコン通信ネットにおいて，専用会議室を設けておりますので，ご利用ください．

●日経MIX／vacs会議
●NIFTY-Serve／VACS Forum（FVACS）

※VACS通信は今回で連載を終了致します．ご愛読いただき，ありがとうございました．（編集部）

アドミラルシステム通信4

Information from Admiral System

いまアドミラルシステムは，AS-DiskOpt(98版)(注：アドミラルシステムディスクオプトですが，「ディスクオプト」と呼んでください)を予定通り11月8日に出荷して，束の間ホッとひと息ついているところです．

さて今月は，これまで書いてきたDOSのディスクドライブ管理の初級・中級編という趣きから離れて，皆さんがあまり知る機会のないであろうソフトウェア開発の現場についてお届けします．

AS-DiskOpt(98版)のβテスト顛末

予定通り出荷に漕ぎつけることができたAS-DiskOptですが，最後の大仕事「βテスト」は――新製品出荷前ではいつものことですが――「そんなバカな!」「ちゃんとチェックしたと言ったじゃあないか!」「見つけたっ!」といった大声が飛び交う騒ぎの連続でした．

AS-DiskOptは，ディスク最適化再配置プログラムという機能の性格上，致命的なトラブルがあった場合には，そのドライブに入っていたファイル内容を破壊してしまうことになります．弊社はこれまで10万本以上をお客様に販売してきました．信じがたいことかもしれませんが，10万人のユーザがいれば10万人の異なったマシン&DOS環境があるというのが実態なのです．

βテストは，この10万人10万色の環境であっても，できる限りトラブルを起こさないようにするために絶対に必要であると考え，実施しています．

蛇足ですが，たとえβテストが完全で，10万人が大丈夫でも，トラブルはいつか起こります．100,001人目では何があるか分からないのがこの世界の常識なのです．なかなか言いにくいことではありますが，できるかぎり開発者が開発に使用した環境，もしくは商品のターゲットとして想定していた環境に近い環境でお使いになることが賢明です．これはソフトウェア開発の現場事情を知る者の共通した本音です．その環境では，まずトラブルはないといってよいでしょう．パソコンの世界でも(?)少数派は圧倒的に不利なのです．できる限り特殊なことは避けることが，安定した環境構築の近道です．

話は元に戻って，日経MIXとNIFTY-Serveの弊社サポート窓口にて，βテストをやっていただける方を募りました．このβテスタの方々にいろいろと試験していただき製品版として出荷できるレベルに達することができたわけです．

もちろんαテストを十分に行って，いろいろな環境内でこのようなトラブルがないことを確認してからβテストにお出ししています．しかし，やはり10万人10万色でした．

弊社内では，正直思いもかけないことでしたが，βテスタの方のドライブ内容が壊れたとことが何度もありました．そのつどご報告いただいたトラブル状況を解析して社内で再現試験を行います．

この再現試験が最も大変な作業です．ある種の組合わせのときにだけ発生する現象であった場合，その組み合わせが特定できるまで，いただいたデータをもとに推測して「アーでもないコーでもない．いや，やっぱりさっきの延長線上だろう……」といった議論を喧喧諤諤しながら，再現するまで，朝から晩までズーとやり続けます．このためのテストチームが社内にはあって，βテスト期間中はこれだけがこのチームの仕事です．

いくらβテスタから不具合の報告をいただいても，このテストチームが再現条件件を突き止めて，開発者に具体的に示せない限り，どんなに優秀な開発者といえども，指摘された不具合は直せません．

テストチームは，開発者が不具合を修正後に多面的なテストを行って，本当に修正されたのか，副作用が他に及んでいないか，という検証を行います．

ひと口にソフトウェア開発といっても，純粋な修正以外にこれだけの手間がかかります．もちろん，これ以外にマニュアル作成やソフトウェアとマニュアルの整合性チェック，といったことはたっぷりとありますが…….

こうしてβテスタの方々から数多くの不具合報告をいただき，社内の再現テストを行って致命的トラブルを徹底的に潰し，10万人10万色の多くの部分は潰せたものと考え，製品出荷GOというサインが出るわけです．

当然のことながら，βテスタの方々の環境で致命的トラブルが起きることはないでしょう．メーカーの勝手な期待としては，弊社内およびβテスタの方々と，ほぼ同じ環境でお使いいただければ，まずトラブルもなく安心してお使いいただけるなあー，と虫のよいことを考えています．

ここでお一人ずつのお名前をお出しすることはできませんが，βテストにご協力いただけた皆様，中にはディスク内容を破壊してしまって申し訳ないこともありましたが，おかげさまで無事出荷することができました．

また，「これはこうした方がよいのでは……」といった建設的なご意見をいただき，製品版に反映させていただいたものもあります．誌面をお借りしまして，厚く御礼申し上げます．

なお弊社では，今後も商品開発の最後のフェーズとして，βテストを実施していきたいと考えています．それは，先に書きましたように，自分の環境を

メーカーである私どもに伝えてテストさせる絶好の機会でもあります.
ご自分の環境を弊社製品に反映させたいとお考えの方，βテストは情報のやり取りが頻繁なために，電子メールが使用できることが必須条件です．弊社では，日経MIXもしくはNIFTY-Serveに公式サポート窓口を設けて，製品のサポートを中心として，ときにはβテストなどを行っております．これからはこの窓口をご覧になるようになさっていただけると幸いです．

■AS-DiskOpt(98版)を始めた本当の理由

出荷し始めたAS-DiskOpt(98版)ですが，これを開発することになった本当の理由をお教えしましょう．
アドミラルシステムは，もとをたどるとソフトウェアパッケージを生業といた会社ではありませんでした．あるひょんなきっかけから，この商売に入ってしまったのです．
そのきっかけは，今は昔となりましたが，東芝のJ-3100というモデルでした．これは当時，仕事でIBM PC/ATを使っていましたが，日本語のAT互換機が，しかも持ち運べるラップトップモデルで登場したのです．
日本で初めての日本語対応AT互換機でしたが，初めての宿命というか，ほとんどサポートする環境やソフトウェアが何もない状態で，卓越したこのマシンを実用的に使うには環境を自前で作るしか方法がない状態でした．
そこで，社内専用ソフトウェアを作成したのですが，ひょんなきっかけからこれを外販することになり，アドミラルシステムはソフトウェアパッケージを開発・販売することになったのです．
話が長くなるので途中省略しますが，このときから今日に至るまでずっと，外販してきたソフトウェアは，社内ニーズを満たすことが開発の発端となっています．言ってみれば「まず社内ニーズありき」なのです．
DOS環境のころには，Newton-XXをこの社内ニーズを満たすために企画し，社外にデビューしました．LANを導入してLANに対応したNewton PRO-XXは社内ニーズを満たすために企画しました．
そして，このAS-DiskOptの場合は，簡単に言えば社内が完全にMS-Windows3.x環境に移行したことが，企画のきっかけとなりました．
すでに何回か書いてきましたし，ご存じの方もいらっしゃるでしょうが，現在弊社内は業務全体がLANに接続したパソコンの上で行うようになっており，基本的にWindows3.x環境がベースとなっています(最近一番Windowsを使えなかったのが前記テストチームでしょう．AS-DiskOpt(98版)はWindows3.1では最適化再配置禁止となっていますので，DOS上が仕事の中心でした)．
Newton PRO-XXは，全体としてみればLAN環境とWindows3.1との親和性が高いといえます．すでに使い込んでいらっしゃる方はお分かりのように，Windows3.1自体のファイル数も相当なものですが，使っている間にどんどんデータファイルが増えていきます．
気がついてみると，それまで十分と思っていたNewton PRO-98付属のディスク最適化再配置プログラムでは「メモリ不足です」というメッセージが出てしまって，ディスクの最適化ができなくなってしまったのです．
これは，AS-DiskOptを出荷できた今だからお話できることですが，ディスクを最適化するのとしないのとでは，マシンの体感速度が圧倒的に違うことは社内の誰もが身を持って知っていたことでしたから，そのメーカーであるだけに，これには頭を抱えました．さっそく開発をスタートさせたのですが，98版は最初から難航しました．
世界標準となっていて，最近は日本でもDOS/V機として勢力を広げてきているAT互換機の世界とは異なり，98の世界では，調べれば調べるほどメモリマネージャがバラバラ．開発者からは「このメモリマネージャは規格を満足しているというが，こんな状態でサポートできるだろうか……?」といった不満(?)というか不安(?)が上がってくる始末．
そうこうするうちに，案の定Newton PRO-XXのユーザから，社内で起きている現象とまったく同じことが起きている，という報告が入るようになってきました．
そのころ，社内ではα版がテストに入り，まずは基本的なバグを潰し，そしていろいろなフェーズを経て，先に書いた最後のβテストへなだれ込んでいったわけです．

●

現在，これを書いている時点で，AS-DiskOpt(DOS/V版)もだいたいメドが立ちました．基本部分は先のAS-DiskOpt(98版)でできあがっていますので，あとは若干のβテストを残すだけとなっています．この号が読まれるころに，AS-DiskOpt(DOS/V版)も日の目を見ているはずなのですが，どうなっていますのでしょうか．

■Newtonデーを定期開催

アドミラルシステムのサポートセンター「ガリオン」(JR川口駅前)では，毎月第4土曜日の午後1時から5時まで，Newtonデーと称して，当社製品のデモや講習などを行っています．もちろん，当社製品とは関係のないパソコンに関するご質問もお受けします．どなたでもご参加できます．参加は無料です．詳細は，ガリオン(TEL 048-259-5111)までお問い合わせください．

(株)アドミラルシステム企画部・黒岩潤司)

(株)アドミラルシステム　営業部
TEL 048-252-4646
FAX 048-252-4657

メテヴェーレス先生の●

ワンポイントテクニック

■適応機種　PC-9801シリーズ
■OS　MS-DOS Ver.3.1以上
■言語　N88-BASIC(86)
QuickBASIC

今月のテーマ

ACTIVE SCREEN MESSAGES

活動的なメッセージ

●著者●　P. Metevelis

はじめに

1. 画面にいくつかのウィンドウが表示されているが，どれがアクティブなウィンドウなのか分からない．
2. 画面にいろいろなメッセージが表示されているが，どれが一番関係の深いものか分からない．

以上のようなことを経験したことはありませんか？

他人の書いたプログラムの場合は仕方のないことですが，自分でプログラムを書く場合は，ユーザがこのような「分からない」という状態に陥らないように設定することができます．実は，そのように設定することはプログラマの責任なのです．

今回は，こうしたことをふまえて活動的なメッセージをテーマにします．活動的なメッセージにより，アクティブウィンドウとアクティブでないウィンドウを区別することができます．また，メッセージの多い画面では，ある特定のメッセージを目立たせることもできます．

活動的なメッセージ

メッセージを目立たせるためには，ブリンク表示や色別表示という方法があります．これらは良い方法で，いろいろなプログラムに使われていますが，そのためユーザがこれらの方法に飽きてしまう場合もあります．また，これらはアクティブウィンドウを示すことができません．とにかく，もうちょっと美術的な方法が欲しいという場合もあるでしょう．そう思うと，以下のようなテクニックがパッと思い浮かびます．

1.メッセージといっしょに動くもの
(moving object associated with message)

- くるくる回るカーソル(twirling cursor)
- 字の上に乗るスプライト(sprite overrider)
- キラキラの火花(sparkles)
- はねまわるボール(bouncing ball)

2.動くメッセージ(moving message)

- ふるえるメッセージ(oscillating message)
- 流れるメッセージ(streaming message)
- またたくメッセージ(blinking message)

3.色が替わるメッセージ(color-changing message)

- 流れる輝きのメッセージ
(message having a moving gleam)
- 流れる色のメッセージ(message in streaming colors)
- きらめく色のメッセージ(message in flashing colors)
- ほのかな色のメッセージ(glimmering message)

4.活動的な枠(active frame)

- きらめく色の枠(frame in flashing colors)
- 回転する色の枠(frame in revolving colors)
- 流れる点線の枠(循環枠)
(frame in flowing dotted lines)

このほかにも，いろいろな方法があることでしょう．ここでの重要なポイントは，画面を面白く，分かりやすくすることでユーザのキーインの間違いが少なくなるだろうということです．

今回のプログラム

活動的なメッセージの例として，四つのプロシージャを私自身の書いたプログラムから選択しました(**リスト1～4**)．**リスト1**と**リスト2**は，大分古くから，8086のためにN88-BASIC(86)で書いておいたもので，そのまま掲載します．あとの二つは386のためにQuickBASICで書いたプログラムです．それぞれの特徴を以下に示します．

① **リスト1**のCREEPER.BASは，ユーザの注目を大きく集めます．このプログラムの示すテクニックは，表示パネルがあって，スプライトが繰り返しにメッセージをそのパネルに運んできます．

プログラムの中で，その活動を行うプロシージャは340行の下の「Creeper Routine」です．行160,210,440やサブルーチン*BLACKGND，*BLUEGNDなどは，デモ用ですので除いても構いません．

なお，このプログラムをQuickBASICに移植するときは，タイミングに気をつけてください．

② **リスト2**のWAIT.BASは，**リスト1**と異なりユーザの注目を大きく集めないで，小さく集めます．これは，キーインを待ちながらメッセージの字の上で流光を送るプロシージャです．その光は細かいものなので，画面の他の部分に邪魔しません．しかし，その活動は十分にアクティブウィンドウや対話的なメッセージを示すことができます．

このプログラムはQuickBASICに移植しやすいものです．ただし，COLOR@の替わりにPRINT 文により色を換えてください(あるいは本誌'90年2月号「QuickBASICでcolor@文を実行する」のCOLORATというプロシージャを使ってください)．

③**リスト3**のSparkle.BASでは，メッセージの文字の周りにランダムに，火花がキラキラします．そのSUB MakeSparklesは，火花を作成する初期化プロシージャです．初期化が完了すると，SUB SparklingWaitはキーインを待ちながら火花を泡のように出したり消したりしていきます．

④**リスト4**のDizzyBox.basのアイデアはPageMakerから借りたものです．これはメッセージの周りを点線の枠が流れます．この活動は必ずユーザの注意を引きますが，ブリンク属性文字よりやさしいです．

画面マスク

DizzyBox.basの枠は，画面マスクの枠としてピッタリです．**リスト5**のDizyMask.basは，はやどりに書かれていますが，それが5ウィンドウの例になります．そのFUNCTION DizzyBox$は入力を受けながら現在ウィンドウの枠に点線を流れさせます．入力が完了されると，↵により，そのウィンドウに

LINE (x1%, y1%)-(x2%, y2%), , B

が実行されます．それにより点線が普通の直線になり，ウィンドウが閉じます．

FUNCTION DizzyBox$ のタイムディレイによってキーの応答はかなり遅いのですが，応答タイムの問題は簡単に直すことができます．そういう直しは，読者の宿題です．

〈参考文献〉

1)田畑純『QuickBASICでcolor@を実行する』TheBASIC '90年2月号

List1 鳥の引っ張るメッセージ
CREEPER.BAS

```
10 '=====================================================================
20 'SAVE"CREEPER.BAS",A       'N88BASIC(86)      86-01-16        P. METEVELIS
30 '---------------------------------------------------------------------
40 ' Program prints an attention-getting message onscreen.
50 ' Timing is for machines with an 8086 CPU. Diagnostics are for demo only.
60 '
70 CLOSE:CLEAR:ON ERROR GOTO *HALT:ON STOP GOSUB *QUIT:STOP ON:DEFINT A-Z
80 WIDTH 80,25:CONSOLE 0,25,0,1,1:COLOR 6,0,0,7:SCREEN 3,0,0,1:CLS 3
90 '
100 REM     <M2$ & M4$ must have same length; likewise for M1$ & M3$>
110 M1$=" Take action! _▲ ":M2$="●▲_ "
120 M3$=" Take action! ‾▼ ":M4$="●▼‾ "
130 MSGTAG$="Press RETURN key "
140 '
150 GOSUB *BLUEGND:GOSUB *TUNNEL
160 LOCATE 20,12,0:PRINT"Diagnostics:"
170 X$="":CTRL=0:CUTOFF=25
180 '
190 '------------------------------<Demo Routine>----------------------------
200  WHILE X$=""
210    LOCATE 34,12:PRINT CTRL
220    GOSUB *CREEP
230    IF CTRL THEN FOR T=1 TO 6800!:NEXT T
240    IF CTRL THEN FOR I=CUTOFF TO 80:GOSUB *TAG:NEXT
250    CTRL=1-CTRL
260  WEND
270  GOSUB *BLACKGND
280 *HALT:LOCATE ,,1:STOP OFF:CLS 3:ON ERROR GOTO 0:END
290 *QUIT:RETURN *HALT
300 *TAG
310  LOCATE 0,0:PRINT LEFT$(STRING$(I,32)+MSGTAG$,80);
320 RETURN
330 '
340 '---------------------------<Creeper Routine>----------------------------
350 *CREEP
360  IF CTRL THEN GOSUB *INIT1 ELSE GOSUB *INIT2
370  MN=0:MX=79-LEN(M$)
380
390  FOR X=A TO B STEP S:X$=INKEY$:IF X$<>"" THEN X=B:GOTO 500
400    IF (CTRL=0)*(X=CUTOFF+10) THEN M$=M3$
410    IF CTRL*(X MOD 5=0)*(X<CUTOFF) THEN M$=M4$
420    IF CTRL*(X MOD 9=0)*(X<CUTOFF) THEN M$=M2$
430    IF CTRL*(X=CUTOFF) THEN M$=M2$:BEEP 1:BEEP 0
440    LOCATE 38,12:PRINT USING "###";X;
450    IF X<MN THEN *HOP ELSE IF MX<X THEN *TROD
460    LOCATE X,0:PRINT M$;:GOTO *JUMP
470   *HOP:LOCATE 0,0:PRINT RIGHT$(M$,LEN(M$)+X);:GOTO *JUMP
480   *TROD:LOCATE X,0:PRINT LEFT$(M$,80-X);
490   *JUMP:FOR T=1 TO 90:NEXT T
500  NEXT
510 RETURN
520 '
530 *INIT1
540  M$=M2$+MSGTAG$:S=-1:A=79:B=-(LEN(M2$))
550 RETURN
560 '
570 *INIT2
580  M$=M1$:S=1:A=-(LEN(M$)):B=79
590 RETURN
600 '
610 '-------------------------<Background Pattern>--------------------------
620 *BLUEGND
630  FOR J=1 TO 3
640    FOR I=10 TO 350 STEP 5
650      LINE(319-I,199-I)-(I+319,I+199),J AND 1,B,&H6666
660    NEXT
670  NEXT J
680 RETURN
690 '
700 *BLACKGND
710  FOR I=10 TO 350 STEP 5
720    LINE(319-I,199-I)-(I+319,I+199),J AND 0,B,&H6666
730  NEXT
740 RETURN
750 '
760 *TUNNEL
770  LINE(0,0)-(639,16),1,BF
780 RETURN
```

List2 待機の微光
WAIT.BAS

```
100 '=====================================================================
110 'SAVE "WAIT.BAS",A       'N88BASIC(86)      86-01-22        P. METEVELIS
120 '---------------------------------------------------------------------
125 '
130 CLEAR:ON ERROR GOTO *TERMINATE:ON STOP GOSUB *INTERCEPT:STOP ON
140 DEFINT A-Z:W=80:WIDTH W,25:CONSOLE 0,25,0,1,1:COLOR 4:CLS 3
150 POSN=15:L4=16
160 '
161 '-----------------------------<Wait Routine>----------------------------
170 *START
179  REM     Sample message
```

```
180 LOCATE POSN,L4:PRINT "Enter ESC to terminate, any other key to continue"
190 I=0:C=6
200 GOSUB *FLUSH
210 WHILE X$=""
220   COLOR@(POSN-1+I,L4)-(POSN-1+I+1,L4),C
230   COLOR@(POSN-1+I-1,L4)-(POSN-1+I,L4),C+2*(C=6):GOSUB *DELAY
240   I=I+1:IF I=50 THEN I=0:C=6+2*(C=6)
250   X$=INKEY$:IF X$=CHR$(27) THEN *TERMINATE
260 WEND
270 GOTO *START                    '<NB this line is for demo purposes only>
280 '
281 '------------------------------<Terminal>--------------------------------
290 *TERMINATE:COLOR 6:CLS 3:LOCATE 0,22:STOP OFF:ON ERROR GOTO 0:END
300 *INTERCEPT:RETURN *TERMINATE
310 '
320 '-----------------------------<Subroutines>------------------------------
330 *DELAY:FOR T=1 TO 300:NEXT T:RETURN             '<1-1300 for 80386 CPU>
340 *FLUSH:X$=INKEY$:WHILE X$<>"":X$=INKEY$:WEND:RETURN
```

List3 待機キラキラのデモ
SPARKLE.BAS

```
'======================================================================
'Sparkle.bas Ver. 2.0; P. Metevelis; 1993-05-15
'----------------------------------------------------------------------
'  Program demonstrates a sparkling wait-procedure.
'----------------------------------------------------------------------

DECLARE SUB application ()
DECLARE SUB MakeSparkles ()
DECLARE SUB SparklingWait (x1%, y1%, x2%, y2%, x$)

REM     Initialization
 REM     Background color for window
 COMMON SHARED wc%
 wc% = 0
 '
 REM     Prepare animation-frames for sparkle sequence
 DIM SHARED pop0%(37), pop1%(37), pop2%(37), pop3%(37)
 CALL MakeSparkles

REM     Run simulated window application
 CALL application

END


'======================================================================
 SUB application
'----------------------------------------------------------------------
'Procedure simulates a window application.
'Input: x$ (key input from wait procedure).

 DO UNTIL UCASE$(x$) = "Q" OR x$ = CHR$(27)
   REM     Simulated application message
   LOCATE 12, 30, 0
   PRINT "Enter RET to do nothing,"
   LOCATE 13, 30, 0
   PRINT "Q to quit this program"

   CALL SparklingWait(224, 172, 420, 212, x$)
 LOOP

END SUB

'======================================================================
 SUB MakeSparkles
'----------------------------------------------------------------------
'Procedure sets up sparkle-animator frames.

 CLS

 REM     Make disintegrator frame
  LINE (0, 0)-(8, 8), wc%, BF
  GET (0, 0)-(8, 8), pop0%

 REM     Make stage-1 frame
  LINE (2, 2)-(4, 4), 7
  LINE (4, 2)-(4, 4), 7
  LINE (6, 2)-(4, 4), 7
  LINE (4, 4)-(2, 4), 7
  LINE (6, 4)-(4, 4), 7
  LINE (2, 6)-(4, 4), 7
  LINE (4, 6)-(4, 4), 7
  LINE (6, 6)-(4, 4), 7
  GET (0, 0)-(8, 8), pop1%

 REM     Make stage-2 frame
  LINE (0, 0)-(8, 8), wc%, BF
  LINE (0, 0)-(2, 2), 7
  LINE (4, 0)-(4, 2), 7
  LINE (8, 0)-(6, 2), 7
  LINE (0, 4)-(2, 4), 7
  LINE (6, 4)-(8, 4), 7
  LINE (2, 6)-(0, 8), 7
  LINE (4, 6)-(4, 8), 7
  LINE (6, 6)-(8, 8), 7
  GET (0, 0)-(8, 8), pop2%
```

```

REM     Make stage-3 frame
 LINE (0, 0)-(8, 8), wc%, BF
 PSET (0, 0), 7
 PSET (4, 0), 7
 PSET (8, 0), 7
 PSET (0, 4), 7
 PSET (8, 4), 7
 PSET (0, 8), 7
 PSET (4, 8), 7
 PSET (8, 8), 7
 GET (0, 0)-(8, 8), pop3%

CLS

END SUB

'=====================================================================
SUB SparklingWait (x1%, y1%, x2%, y2%, x$)
'---------------------------------------------------------------------
'Procedure sparkles in a window(or specified area)while awaiting key input
'Input: x1%, y1%, x2%, y2% (boundaries of sparkling area).
'Output: x$ (key value).

REM     Initialize loop
 xq% = x1% + FIX(RND * (x2% - x1% - 8))
 yq% = y1% + FIX(RND * (y2% - y1% - 8))

REM    Sparkle-wait loop
 DO
  REM    Sparkle sequence initialization
  xqbak% = xq%: yqbak% = yq%
  xq% = x1% + FIX(RND * (x2% - x1% - 8))
  yq% = y1% + FIX(RND * (y2% - y1% - 8))
  '
  REM     Sparkle sequence
  PUT (xqbak%, yqbak%), pop0%, PSET     '<disintegrate previous sparkle>
  PUT (xq%, yq%), pop1%, PSET                        '<sparkle stage 1>
  FOR i% = 1 TO 1100: x$ = x$: NEXT i%               '<microdelay loop>
  PUT (xq%, yq%), pop2%, PSET                        '<sparkle stage 2>
  FOR i% = 1 TO 1100: x$ = x$: NEXT i%               '<microdelay loop>
  PUT (xq%, yq%), pop3%, PSET                        '<sparkle stage 3>
  '
  REM     Key input routine
  x$ = INKEY$
  IF x$ <> "" THEN
    PUT (xq%, yq%), pop0%, PSET       '<disintegrate remaining sparkle>
    EXIT DO
  END IF
 LOOP

END SUB
```

List4 よっぱらった枠(循環枠)
DIZZYBOX.BAS

```
'=====================================================================
'DizzyBox.bas; Ver. 2; QB4.2; P.Metevelis; 1990-11-07
'---------------------------------------------------------------------

CLS

REM     Sample message
 LOCATE 8, 17: PRINT "This"
 LOCATE 9, 18: PRINT "is"
 LOCATE 10, 19: PRINT "a"
 LOCATE 11, 16: PRINT "message"

REM     Make box dizzy
 DO
   LINE (100, 100)-(200, 200), 0, B, &H9999
   '
   LINE (100, 100)-(200, 200), 7, B, &H3333
   GOSUB SlowDown:
   LINE (100, 100)-(200, 200), 0, B, &H3333
   '
   LINE (100, 100)-(200, 200), 7, B, &H6666
   GOSUB SlowDown:
   LINE (100, 100)-(200, 200), 0, B, &H6666
   '
   LINE (100, 100)-(200, 200), 7, B, &HCCCC
   GOSUB SlowDown:
   LINE (100, 100)-(200, 200), 0, B, &HCCCC
   '
   LINE (100, 100)-(200, 200), 7, B, &H9999
   GOSUB SlowDown:
 LOOP WHILE INKEY$ = ""

END

'---------------------------------------------------------------------
SlowDown:
  FOR delay = 1 TO 500: NEXT                              '<microdelay>
RETURN
```

List5 循環枠付きの画面マスク
DIZZYMASK.BAS

```
'=====================================================================
'DizyMask.bas; Ver. 1; QB4.2; P.Metevelis; 1993-05-11
'--------------------------------------------------------------------
'Program demonstrates the usefulness of the dizzybox in a screen mask.
'For the demonstration, string values are inputted into the array record$
'Note that this program is roughly written.

DECLARE FUNCTION DizzyBox$ (x1%, x2%, y1%, y2%, l%, c%, wdlen%, retCode%)
DECLARE SUB nextBox (n%, x1%, y1%, x2%, y2%, l%, c%, wdlen%)
DECLARE SUB clearWins ()

REM     Initialization and patch area
 rmax% = 15
 DIM record$(5, rmax%)

REM     Prepare screen
 CLS
 PRINT "RET for next item"
 PRINT "ESC to terminate"
 PRINT "    records"

REM     Input data
 rn% = 1
 LOCATE 3, 1: PRINT USING "###"; rn%
 DO
   n% = n% + 1
   CALL nextBox(n%, x1%, y1%, x2%, y2%, l%, c%, wdlen%)
   record$(n%,rn%)=DizzyBox$(x1%, x2%, y1%, y2%, l%, c%, wdlen%,retCode%)
   IF retCode% = 2 THEN EXIT DO
   IF n% = 5 THEN
     n% = 0
     rn% = rn% + 1
     IF rn% > rmax% THEN EXIT DO ELSE LOCATE 3, 1: PRINT USING "###"; rn%
     CALL clearWins
   END IF
 LOOP

REM     Output results
 CLS
 FOR j% = 1 TO rmax%
   FOR i% = 1 TO 5
     PRINT record$(i%, j%); "; ";
   NEXT
   PRINT
 NEXT

END




'=====================================================================
 SUB clearWins
'--------------------------------------------------------------------
'Procedure clears input windows. Values must correspond to l%, c%, wdlen%
'respectively in the nextBox procedure.  They would best be parametrized.

 LOCATE 7, 14: PRINT SPACE$(11)
 LOCATE 9, 14: PRINT SPACE$(11)
 LOCATE 11, 14: PRINT SPACE$(11)
 LOCATE 13, 14: PRINT SPACE$(11)
 LOCATE 15, 14: PRINT SPACE$(11)

END SUB

'=====================================================================
 FUNCTION DizzyBox$ (x1%, x2%, y1%, y2%, l%, c%, wdlen%, retCode%)
'--------------------------------------------------------------------
'Input: x1%, x2%, y1%, y2%, l%, c%, wdlen%.
'Output: retCode%.
'Return codes: 1 = entry complete (not presently used); 2 = end program.

 DO
   REM     Make box dizzy
   LINE (x1%, y1%)-(x2%, y2%), 0, B, &H9999
   LINE (x1%, y1%)-(x2%, y2%), 7, B, &H3333
   GOSUB SlowDown:
   LINE (x1%, y1%)-(x2%, y2%), 0, B, &H3333
   LINE (x1%, y1%)-(x2%, y2%), 7, B, &H6666
   GOSUB SlowDown:
   LINE (x1%, y1%)-(x2%, y2%), 0, B, &H6666
   LINE (x1%, y1%)-(x2%, y2%), 7, B, &HCCCC
   GOSUB SlowDown:
   LINE (x1%, y1%)-(x2%, y2%), 0, B, &HCCCC
   LINE (x1%, y1%)-(x2%, y2%), 7, B, &H9999
   GOSUB SlowDown:

   REM     Simple input routine
   retCode% = 0
   x$ = INKEY$
   IF x$ <> "" THEN
     IF x$ = CHR$(13) OR x$ = CHR$(27) THEN
       LINE (x1%, y1%)-(x2%, y2%), , B
       IF x$ = CHR$(27) THEN retCode% = 2 ELSE retCode% = 1
       EXIT DO
     ELSEIF (ASC(x$)>=&H20) AND (ASC(x$)<=&H7A) AND LEN(txt$)<wdlen% THEN
```

```
        txt$ = txt$ + x$
    ELSEIF x$ = CHR$(8) AND LEN(txt$) > 0 THEN
        txt$ = LEFT$(txt$, LEN(txt$) - 1)
    END IF
    LOCATE l%, c%: PRINT txt$; " ";
  END IF
LOOP WHILE x$ <> CHR$(13) AND x$ <> CHR$(27)

REM     Results
  DizzyBox$ = txt$
EXIT FUNCTION

SlowDown:
  FOR delay = 1 TO 500: NEXT
RETURN

END FUNCTION


'==========================================================
SUB nextBox (n%, x1%, y1%, x2%, y2%, l%, c%, wdlen%)
'----------------------------------------------------------
'Procedure produces screen coordinates for successive input windows.
'Input: n% (all other variables are output).

SELECT CASE n%
  CASE 1
    x1% = 100: y1% = 94
    x2% = 200: y2% = 116
    l% = 7: c% = 14: wdlen% = 11
  CASE 2
    x1% = 100: y1% = 126
    x2% = 200: y2% = 148
    l% = 9: c% = 14: wdlen% = 11
  CASE 3
    x1% = 100: y1% = 158
    x2% = 200: y2% = 181
    l% = 11: c% = 14: wdlen% = 11
  CASE 4
    x1% = 100: y1% = 190
    x2% = 200: y2% = 213
    l% = 13: c% = 14: wdlen% = 11
  CASE 5
    x1% = 100: y1% = 221
    x2% = 200: y2% = 245
    l% = 15: c% = 14: wdlen% = 11
END SELECT

END SUB
```

コ・ン・ピ・ュ・ー・タ COMPUTER ウ・イ・ル・ス VIRUS カ・ル・テ KARTE

㉛ *Changsha*

“Changsha”は，平成5年10月11日に大阪府で発見された中国産のウイルスです．

解析の結果，IBM-PC/互換機用に作成された，COM，EXEファイルに感染した状態で流通するウイルスであることが分かりました．

“Changsha”に感染したプログラムを実行すると，まず自分自身がメモリに常駐しているかチェックをします．常駐していない場合は，Cドライブのハードディスク，マスターブートレコードに感染し，メモリに常駐してプログラムの実行とファイルの検索を監視します．ハードディスクがない場合でもウイルスは，メモリに常駐します．また，ハードディスクへ感染する前に，すでに自分自身が感染しているかチェックを行っていますので，ハードディスクへ重複して感染することはありません．

“Changsha”がメモリに常駐した状態で，プログラムの実行や，ワクチンプログラムなどを使用してファイル検索を行うと，実行したプログラムや，ファイル検索で発見した拡張子“.??M”と“.??E”ファイルにウイルスが感染します(?は省略しない限り，何が入ってもよい)．このように，感染するファイルを拡張子のみで決定しているため，拡張子“.ASM”などのファイルに感染した場合は，感染によってファイルを破壊します．また，ファイル名のチェックを感染前に行っていますので，ルートディレクトリにある先頭が“CO”で始まるファイルと“IB”で始まるファイルには感染しません．

このウイルスに感染したハードディスクから起動した場合も，同様にウイルスがメモリに常駐し，拡張子“.??M”と“.??E”ファイルにウイルスが感染します．

このウイルスには，2種類の発病があります．まず，一つめの発病ですが，毎年5月4日にプログラムの実行，またはファイルの検索を行うと発病し，実行または検索したファイルを削除して，以下のメッセージを画面の左隅のほぼ中央に背景を赤にして表示し，ベル音を「ピッ」と一つ鳴らします．

```
'XqR:                                    '
'                                        '
'  Wherever, I love you Forever and ever !'
'  The beautiful memory for ours  in that '
'summer time has been recorded in the Com-'
'puter history.                          '
'  Bon voyage, My dear XqR !             '
'                                        '
'            Yours 05121991 in our Home.  '
```

もう一つの発病は，毎週日曜日に起こります．日曜日に作業をしていると，突然[Ctrl]キーと[Num Lock]キーと[Caps Lock]キーが押されたままの状態になり，キー入力が妨害されます．発病のタイミングはDOSのファンクションコールを36,964回実行されたときとなるため不定で，いつ発病するか分かりません．

ウイルス名称	Changsha		
ウイルス俗称	なし		
起源	1992年12月　中国		
国内発見日	1993年10月11日　大阪		
ステータス	NEC	IBM	富士通
▶感染力	■■■□□	■■■□□	□□□□□
▶露見しにくさ	■■■□□	■■□□□	□□□□□
▶被害	■■■■□	■■■■□	□□□□□
タイプ	MCB内常駐型(感染ファイル実行時) LMB常駐型(HD，MBR感染，HD BOOT時) 実行時汚染型(感染ファイル実行時HD，MBR汚染) 実行時感染型 ファイル検索感染型 COM型実行ファイル後部寄生型 EXE型実行ファイル後部寄生型 BOOT寄生型(HD，MBR) ステルス型(HD，MBR) 暗号(メッセージ)		
ウイルス長	メモリ常駐　3,328バイト COM型実行ファイル　3,072バイト増加 EXE型実行ファイル　3,090～3,105バイト増加 HD内MBR　512+2,560バイト		
OS	MS-DOS，IBM-DOS		

ウイルスがステップトレース割り込み(INT 01H)とブレークポイント割り込み(INT 03H)を破壊しているため，デバッガなどのINT 01H，INT 03Hを使用しているプログラムが暴走します．

このウイルスがメモリに常駐している状態では，DOSのファンクションコール割り込み(INT 21H)の書き換えができなくなるため，INT 21Hを利用している常駐プログラムが正常に動作しなくなります．また，ハードディスクのマスターブートレコードを読み書きした場合は，正しいマスターブートレコードを利用者に読み書きさせるステルス処理を行っています．

●NECのPC-98シリーズの場合

ウイルスは，ほぼIBM-PC/互換機と同様の動作をします．IBM-PC/互換機との動作の違う点は，ハードディスクへの感染がない点，5月4日の発病でメッセージが表示されない点，日曜日の発病がない点，INT 21Hを利用している常駐プログラムでも正常に動作する点です．

●富士通のFM-Rシリーズの場合

このウイルスに感染したファイルを実行すると「不法な割り込み13H番が実行されました」と表示され，マシンが停止するため，ウイルスの感染は広がりません．

●感染確認方法

感染の恐れのないDOSのシステムディスクからマシンを起動します．このウイルスに感染しているファイルは，ファイルサイズがCOMファイルで3,072バイト，EXEファイルで3,090～3,105バイト増加していますので，マスターディスクに保存されているプログラムのファイルサイズと使用しているディスクのプログラムのファイルサイズを比較して差違が認められた場合，このウイルスに感染している可能性があります．

また，このウイルスに感染したファイルの中身をDUMPコマンドなどでのぞくと，以下の文字列が確認できます．

```
'Welcome!'
'Auto-Copy Deluxe R3.00'
'(C)Copyright 1991. Mr. YaQi. Changsha China'
'No one can Beyond me!'

'Invalid partition table'
'Error loading operating system'
'Missing operating system'

'New Century of Computer Now!'
```

●対策方法

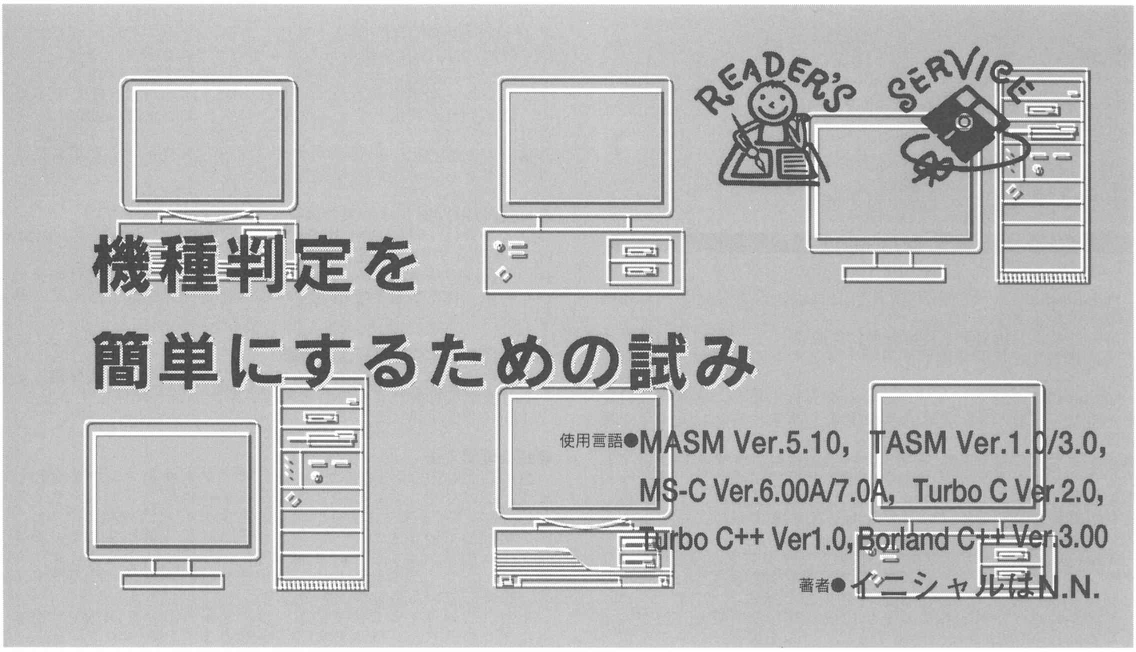
まず，感染の恐れのないDOSのシステムディスクからマシンを起動し，FDISKコマンドの/MBRオプションを使用してハードディスクのウイルスを駆除します．次いで，このウイルスに感染しているファイルをDELコマンドで削除し，マスターディスクからの再インストールまたは，バックアップからのリストアで復旧が可能です．ただし，FDISKコマンドの/MBRオプションはDOSによって使用できないことがありますので，注意が必要です．

●追加パラメータファイル

```
NAME    CHANGSHA
DATA    0 B4H 54HBBH 64H 90HCDH 21H 3DH 64H 90H 75H
DATA    108H 81HC7H 80H 00H 8BH 05HA3H 00H 01H 8BH 45H 02HA3H 02H 01H
8AH
DATA    118H 45H 04HA2H 04H 01H
DATA    2E0HB8H 02H 42H 31HC9H 31HD2HCDH 21HA3H 84H 01HB4H 40HB9H 00H
DATA    2F0H 0CHBAH 00H 01HCDH 21H
DATA    3B3HBFH 40H 01HB8H 02H 42HB9H 00H 00HBAH 00H 00HCDH 21H 89H
45H
DATA    3C3H 2CH 89H 55H 2EH
DATA    3EEH 8BH 45H 16H 89H 45H 24H 8BH 45H 14H 89H 45H 26H 8BH 45H
0EH 89H
DATA    3FEH 45H 28H 8BH 45H 10H 89H 45H 2AH
DATA    464HB4H 2AHCDH 21H 81HFAH 04H 05H 74H 08H
RAM     T,0200H
COM     P,<0E8H>,3
EXE     P,<0E8H>,3

NAME    CHANGSHA(BOOT)
DATA    07FHA1H 13H 04H 2DH 04H 00HA3H 13H 04H
DATA    073HA1H 4CH 00HA3H 62H 7CHA1H 4EH 00HA3H 64H 7CH
DATA    0D6HB8H 01H 02HBBH 00H 7CHB9H 02H 00HBAH 80H 00HCDH 13H
72HF0H
DATA    0E6HEAH 00H 7CH 00H 00H
IPL     T,0
```

㈱ジェード　コンピュータウイルス対策室
〒105　東京都港区虎ノ門3-22-11豊澄ビル2F
㊋03-3435-1348　FAX03-3435-1349
〒420　静岡県静岡市常磐町3-6-11
㊋054-252-0085　FAX054-221-0282



機種判定を簡単にするための試み

使用言語●MASM Ver.5.10，TASM Ver.1.0/3.0，MS-C Ver.6.00A/7.0A，Turbo C Ver.2.0，Turbo C++ Ver1.0，Borland C++ Ver.3.00

著者●イニシャルはN.N.

◆はじめに

DOS/Vの普及に伴って，各メーカーのいろいろなパソコンを触らざるを得ない状況に置かれている人も少なくないでしょう．自作ユーティリティを作る際にはせっかく作るのですから，自分が使う可能性のあるパソコンではすべて動作させたいものですが，自動判定をしようとすると機種判定がわりと面倒です．今回は機種判定を楽にするためのひとつの試みを紹介します．

◆国内で使用されているMS-DOSパソコンの種類

国内で使用されているMS-DOSパソコンをざっとあげてみると，

- PC-9801シリーズ
- PC-9821シリーズ
- EPSON PC-286/PC-386/PC-486
- J-3100シリーズ
- AXマシン
- DOS/Vマシン
- FMR
- FM-TOWNS

などを始めとしていろいろとあります．これらを全部使っている人はいないでしょうが，数種類のパソコンを使っている人は少なくないでしょう．今回は，これらの中から筆者が使用したことのあるパソコンの機種判別を簡単にするためのひとつの試みを紹介します．

◆MS-DOSのOEMのシリアル番号

筆者が最初に使用したDOSマシンは初代PC-9801で，以降しばらくはPC-9801シリーズを使用していました．

■表1■MS-DOSのバージョン番号の取得

DOSファンクション30H MS-DOSのバージョン番号の取得
引　数
AH = 30H INT 21H DOS 5.0では AH = 30H AL = 0 : BHにOEMシリアル番号 = 1 : BHにバージョンフラグ
返り値
AL = バージョン番号の整数部 AH = バージョン番号の小数部 BH = OEMのシリアル番号/バージョンフラグ バージョンフラグ = 08H : ROM DOS BL:CX = 24ビットのユーザーシリアル番号
機　能
MS-DOSのバージョン番号を返す．BHレジスタのOEMのシリアル番号である程度機種判別が可能である． 筆者が使用した経験のあるMS-DOSでは次のようなOEMのシリアル番号となっている． "IBM DOSバージョンJ4.0/V, J5.0/V" :00H "PC-9800シリーズMS-DOS 2.11, 3.1, 3.3, 5.0" :FFH "J-3100シリーズ MS-DOS 2.11, 3.1" :29H シャープAX386用MS-DOS 3.2 :16H MS-DOS 5.0/V :FFH DR DOS 6.0/V :00H "FM-TOWNS MS-DOS 3.1, 5.0" :FFH ただし，AXマシンは同じマシンであってもDOSの日付によりOEMのシリアル番号が異なるものがあるため注意すること．また、MS-DOS 5.0/VもOEM先により異なる可能性がある．

しかし，初代J-3100が小型で結構使えるものだったので，J-3100も並行して使用するようになりました．このとき，困ったのが機種依存するユーティリティです．機種ごとにソースを条件アセンブルや条件コンパイルを行って対応するという方法もありましたが，実行ファイルのバージョン管理が面倒になるため，できれば自動判定を行いたいと思ってMS-DOSのマニュアルと睨めっこしてみました．

そのときに目に止まったのがDOSファンクション30HでBHレジスタに返されるOEMのシリアル番号でした．OEMのシリアル番号はPC-9800シリーズではFFHでJ-3100シリーズでは29Hと異なっていたため，自作ユーティリティではこれで機種判別を行うようになりました(**表1**)．

当分はこれでよかったのですが，その後AXマシンも使うようになるとOEMのシリアル番号が同じメーカーの同じ型式のマシンでもバラバラで統一性がなく，うまくいかなくなりました．そのあと，分かった事情ではOEMのシリアル番号はIBMが00HでMS-DOSがFFHとなっており，MS-DOSのOEM先によりFFHを変更したり，そのままになっていたりといった感じになっていました．

◆機種判別

そこで，まじめに機種判別を行ってみることにしました．その結果が，**リスト1**の後半部分です．機種判別の詳細は**参考文献1〜3**を参照していただくとして以下に簡単に説明します．

◇IBM PC互換機の判定

IBM PC互換機ではBIOSシステム変数がある程度公開されているため，次のBIOSシステム変数を調べています(**表2**)．

①キーボードバッファ先頭ポインタと末尾ポインタ
②RS-232Cチャンネル0のポートアドレスが3f8Hまたは2f8Hで，なおかつプリンタチャンネル0のポートアドレスが378Hまたは278Hまたは3bcH

①または②のOR条件で判定しています．安全のため，どちらもBIOSシステム変数を2カ所確認することになりますから，他機種をIBM PC互換機と判定する可能性はまずないと思います．

DOS/V，J-3100，AXの判定は呼び出しても他に影響を与えない機種固有のROM BIOSを呼び出して，そのファンクションが正常な値を返すかどうかで判定します．

◇DOS/Vマシンの判定

DOS/Vマシンの判定はフォントの読み取り機能のアドレスの取得(**表3**)を使用しています．日本語モードと英語モードの判定はDOSファンクション6300H(**表4**)で行います．

◇AXマシンの判定

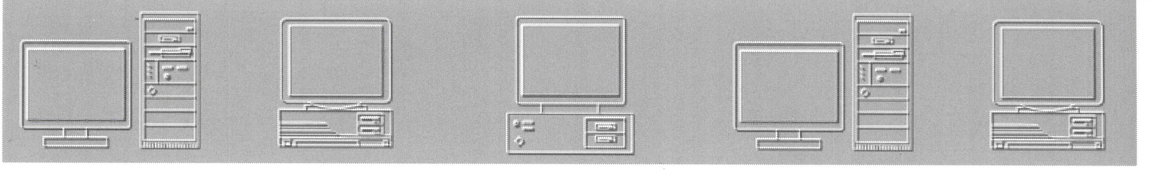
AXマシンの判定はビデオBIOS 5001Hを使用しており，日本語モードと英語モードとの判定も一緒にしています．AXマシンではビデオBIOS，キーボードBIOS，プリンタBIOSで別々に日本語モードと英語モードを切り替えることができますが，簡単にするためビデオBIOSだけで判定しています(**表5**)．

◇J-3100シリーズの判定

J-3100シリーズの判定はビデオBIOS 88H(外字フォントテーブルアドレスの取得)で行っています．日本語モードと英語モードの判定はビデオモードで行っています(**表6**)．

■表2■機種判別に使えるAT互換機のBIOSシステム変数

アドレス	サイズ	内容
0040:0000H	4ワード	RS-232C (0〜3)ポートアドレス
0040:0008H	4ワード	プリンタ(0〜3)ポートアドレス (PS/2では0〜2で40:000EHは予約域)
0040:001AH	1ワード	キーボードバッファ開始ポインタ(読み出し位置)
0040:001CH	1ワード	キーボードバッファ終了ポインタ(書き込み位置)
0040:001EH	16ワード	キーボードバッファ
0040:004AH	1ワード	画面の1行当たりの文字数
0040:0080H	1ワード	キーボードバッファ先頭ポインタ
0040:0082H	1ワード	キーボードバッファ末尾ポインタ
0040:0084H	1バイト	1画面の行数−1



◇PC-9801シリーズの判定

PC-9801シリーズかどうかの判定はPC-9801シリーズのシステム変数領域を調べています.

①**表7**のキーボードバッファのポインタが次のようになっていることを調べます. キーボードバッファのポインタには次の関係があります.

キーボードバッファ次読み出しアドレス+キーボードバッファ格納カウント数×2=キーボードバッファ次格納アドレス

②**表7**のスクロール行数をファンクションキー内容の表示/未表示と20行モード/25行モードでそれぞれ場合分けをして調べ, ①または②のOR条件で判定しています. キーバッファを拡張してなおかつ, 本誌でお馴染みの30行BIOSを使用すると判定を間違えてしまいますが, 筆者の環境では両方を使用することはないため, これ以上の判定は止めています.

◇FM-TOWNSの判定

FM-TOWNSでは**表8**のCPU識別レジスタを調べてい

■表3■フォントの読み取り/書き込みのアドレス取得(DOS/V)

フォントの読み取り/書き込み機能のアドレスの取得

引　数

AH　= 50H
AL　= 00H :フォントの読み取り機能のアドレスの取得
　　= 01H :フォントの書き込み機能のアドレスの取得
BH =フォントの種類
　　ビット0 = 0 : 1バイト文字セット
　　ビット0 = 1 : 2バイト文字セット
　　ビット1～7　:予約済
BL = 0 :予約済
"(DH,DL)=フォントサイズ(文字幅, 文字高さ)"
BP　=コードページ
　　= 0 :コードページの省略値
INT 15H

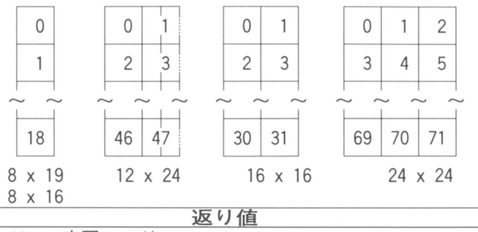
返り値キャリー= 0 :正常終了
　　AH = 00H
　　ES:BX =指定した機能のアドレス
　キャリー= 1 :異常終了
　　AH =完了コード
　　　= 01H :無効なフォントの種類(BH :無効)
　　　= 02H : BLが無効
　　　= 03H :無効なフォントサイズ(DX :無効)
　　　= 04H :無効なコードページ(BP :無効)
　　　= 86H :本機能が未サポート

機　能

文字フォントパターンの読み取りと書き込みを行うためのアドレスを返す. このアドレスを次のような形式で呼び出す.

入　力

CX =文字コード
　"半角コード: CH = 0, CL = ANKコード"
　全角コード:CH =第一バイト, CL =第二バイト
　　　　ES:SI =文字イメージバッファへのポインタ



返り値

AL　= 完了コード
　　= 0 :正常終了
　　= 5 :無効な文字コード(CX :無効)
　　= 6 :指定されたフォントは読み取り専用で書き込めない

■表4■DBCSベクタ情報の取得

DOSファンクション6300H DBCSベクタ情報の取得(DOS3.2以降)

引　数

AX = 6300H
INT 21H

返　り　値

キャリー= 1 :エラー (AX =エラーコード)
キャリー= 0 :成功
　DS:SI = DBCSベクタテーブルアドレス
　　"+0, +1"　DBCSベクタ1開始バイト範囲
　　"+2, +3"　DBCSベクタ2開始バイト範囲
　　......
　　"+?,+?+1"
　　"0, 0"　　ターミネータ
注)①ALレジスタでエラーの有無を示すDOSバージョンもあります.
　AL　= エラーコード
　　= 0　　:成功(DS:SI = DBCSベクタテーブルアドレス)
　　= FFH　:未サポート
注)②CS:IPレジスタとSS:SPレジスタを除く他のレジスタを破壊してしまうDOSバージョンもあります.

機　能

DBCS(2バイト文字セット)ベクタテーブルアドレスを返す. DBCSベクタは2バイト文字セットの開始バイトの範囲を示しており, 日本語では次のようになっています.

"DS:SI+0= 81H, 9FH"　　DBCSベクタ1
"　　+2 = E0H, FCH"　　DBCSベクタ2
"　　+4 = 00H, 00H"　　ターミネータ

これは日本語の漢字コードの第一バイトが81H～9FHまたはE0H～FCHの範囲であることを示しています.
英語バージョンではDOS4.0以降で本ファンクションをサポートしており,

"DS:SI +0 =00H,00H"　　ターミネータ

となります(ただし, ALレジスタでエラーの有無を示す).

■表5■日本語モードと英語モードの取得(AXマシン)

ビデオBIOS 5001H日本語モードと英語モードの取得

引　数

AH　= 50H
AL　= 01H
INT 10H

返　り　値

AL　=ステータス
　　0 :正しく読み取った
　　2 :その他のエラー
BX　=カントリーコード
　　1 :英語
　　81 :日本語

機　能

AXマシンのビデオBIOSの動作モードが日本語モードか英語モードかを返す.

ます．IBM PC互換機やPC-9801シリーズではI/Oポート30H，31Hはバイトポートとなっており，ワードで読み取ると上位バイトがFFHとなるはずです．このCPU識別レジスタの値が

ビット7〜3　＝00H
ビット15〜8 ≠FFH

であることをチェックしています．また，FM-TOWNSのI/Oポート20EHはドライブスイッチレジスタでビット0にのみ機能が割り付けてあり，ビット7〜1は0と規定されているため，これもチェックします．

◆再びMS-DOSのOEMのシリアル番号

上記の機種判別は特殊な使い方をしていない限り，うまく自動判定が可能ですが，次のような欠点があります．

①機種判別処理がわりと大きい(リスト1では約400行).

■表6■外字フォントテーブルアドレスの取得と設定(J-3100シリーズ)

ビデオBIOS 88H　外字フォントテーブルアドレスの取得と設定
引数
AH = 88H ES:BX =外字フォントテーブルの起点アドレス(設定) = 0:0(取得) INT 10H
返り値
ES:BX =外字フォントテーブルの起点アドレス (ES:BX = 0:0を指定した場合)
機能
外字フォントテーブルアドレスの取得と設定を行う．

■表7■PC-9801シリーズMS-DOSシステム変数領域

アドレス	バイト	内　容
0:0502H	32	キーボードバッファ
0:0524H	2	キーボードバッファ次読み出しアドレス
0:0526H	2	キーボードバッファ次格納アドレス
0:0528H	1	キーボードバッファ格納カウント数
60:0110H	1	現在の行位置
60:0111H	1	=0:最下行にファンクションキー未表示 =1:最下行にファンクションキー表示
60:0112H	1	スクロール行数 ファンクションキー内容 （表示／未表示）20行モード：12H／13H，25行モード：17H／18H
60:0113H	1	25行モード =0:20行モード =1:25行モード
60:011CH	1	現在の桁位置

これは機種判別処理をライブラリ化してしまえば，実行ファイルが少し大きくなるだけで特に支障はないかもしれません．

②上記機種判別処理で判定できないマシンに対応するには機種判別処理を変えてすべてのプログラムをリンクし直さないといけない．

IBM PC互換機はJ-3100，AXマシンに続いてDOS/Vが登場し，これ以上の新種は登場しないと思いますが，特殊なデバイスドライバやTSRプログラムを組み込んでいると機種判別を間違える可能性があります．このときに機種判別処理を改良すると，すべてのプログラムをリンクし直さないといけないので面倒です．

これを避けるには機種判別をデバイスドライバかTSRプログラムにすればよいのですが，筆者はOEMのシリアル番号を機種判別に利用していた関係でこれを拡張することにしました．

◆MS-DOS OEMのシリアル番号検査ユーティリティの使用法

使用法1.：device=chgoem.exe [-N<シリアル番号>]

■表8■FM-TOWNS CPU識別レジスタ

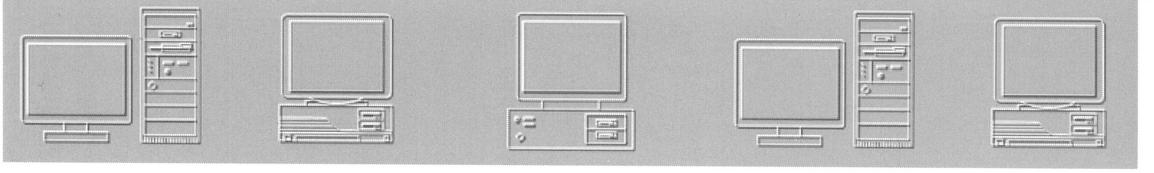
	I/Oアドレス31H								I/Oアドレス30H							
機種	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
新FM-TOWNS	0	0	0	0	0	0	1	0	0	0	0	0	0	CPU種別		
FM-TOWNS	0	0	0	0	0	0	0	1	0	0	0	0	0	CPU種別		
FMR-60/50	不定								1	1	1	1	1	CPU種別		
FMR-50S	不定								1	1	1	0	1	CPU種別		
FMR-70	不定								1	1	1	1	0	CPU種別		

注)
新FM-TOWNS
モデル1F，モデル1H，
モデル2F，モデル2H
FM-TOWNS
モデル1，モデル2

CPU判別	2	1	0
80286	0	0	0
80386	0	0	1
予約済み	×	1	×

■表9■OEMのシリアル番号

機種名	変更値	オリジナル値
PC-9800シリーズ	FFH	←
IBM DOS/V	00H	←
MS-DOS/V	00H	FFH
J-3100シリーズ	29H	←
AXマシン	16H	いろいろ
その他のIBM PC互換機	69H（‘i’）	いろいろ
FM-TOWNS	D4H（80H+‘T’）	FFH
不明	55H（‘U’）	いろいろ



使用法2.：chgoem [-N<シリアル番号>]

本ユーティリティはCONFIG.SYSにデバイスドライバとして登録することと，コマンドラインからEXEファイルとして起動することのどちらも可能です．CONFIG.SYSにデバイスドライバとして登録しても初期化ルーチンを実行するだけで常駐しません．また，コマンドラインから実行しても常駐しません．<シリアル番号>が指定されると指定のOEMのシリアル番号に変更します．スイッチ-Nだけで<シリアル番号>を省略すると**表9**のように変更します．

機種判別を行うデバイスドライバを使用する場合には，そのデバイス行より前に本ユーティリティをデバイスドライバとして登録してください．ただし，デバイスドライバとして登録する際には，環境が整った後に実行させてください．たとえば，MS-DOS 5.0であればHIMEM.SYSとEMM386.SYSを組み込んだ後，DOS/Vの場合はさらに$FONT.SYSと$DISP.SYSを組み込んだ後です．EXEファイルとして実行させる場合はAUTOEXEC.BATの最初で行えばよいと思います．

◆MS-DOS OEMのシリアル番号検査ユーティリティの説明

本ユーティリティは常駐しないEXE型のデバイスドライバの形式をとっています．EXE型のデバイスドライバに関しては**参考文献4**を参照してください．以下ではポイントを簡単に説明します．

上述の方法で機種判定を行ってセットするOEMのシリアル番号を決め，**リスト3～リスト6**に示すDOSファンクション30H部分の該当コードを調べて直接パッチ当てを行います．該当部分はMSDOS.SYSの部分なので，マイナーバージョンや機種が変わってもコードが移動するだけで大きな変更はないと思います．MS-DOS 5.0ではDOSがHMAにロードされているかどうかを調べてパッチ当てを行う個所を変えているため，DOS=HIGH，DOS=LOWのどちらでも正常に動作します．

リスト1ではBHレジスタだけを更新していますが，BL:CXレジスタのユーザシリアル番号はほとんどのDOSで使用されていないようなので，これらにもパッチを当ててより細かい情報を設定するというのもおもしろいかもしれません．たとえば，

日本語モードと英語モードの区別
PC-9821とPC-9801の区別
IBM DOS/V，MS-DOS/V，DR DOS/Vの区別
EPSONマシンでPC-9801シリーズMS-DOS
PC-9801シリーズでEPSON MS-DOS
J-3100シリーズのDOS/V
AXマシンのDOS/V

などや画面の行数と桁数など候補はいろいろとあります．本来は常駐プログラムにしてOEMのシリアル番号が対応済かどうかのステータスを返せるようにするのが安全ですが，極力手を抜いたため現バージョンではパッチ当てだけにとどめています．

動作確認は次の機種とMS-DOSのバージョンで行っています．

PC-9801シリーズMS-DOS 3.1, 3.3B, 5.0A
EPSON MS-DOS 4.0
IBM DOS J5.0/V
MS-DOS 5.0/V
FM-TOWNS MS-DOS 3.1, 5.0
PC-DOS J6.1/V Alpha版

◆MS-DOS OEMのシリアル番号検査ユーティリティのコンパイル

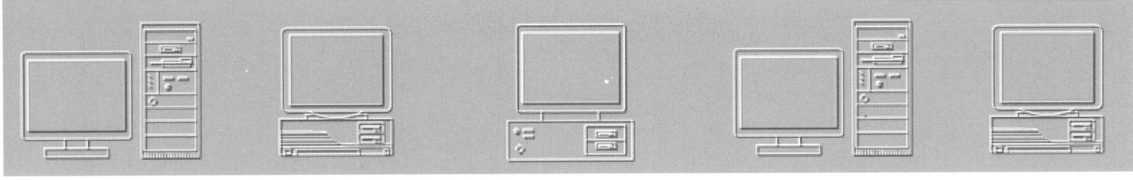
リスト1，**リスト2**のアセンブル，コンパイルは次のようにして行ってください．アセンブラはMASM 5.10，TASM 1.0，TASM 3.0で確認しています．ifdefによるコンパイラ指定を受け付けるアセンブラであればうまくいくはずです．

(1) MS-C 6.00A/7.0A

```
masm /ML devhead;
cl /Ot /J /Gs /W3 /Zp /Fechgoem devhead.obj
  chgoem.c /link /st:0x2800 /cp:0x500
```

(2) Turbo C 2.0

```
tasm /ML devhead;
tcc -c -w chgoem.c
```



```
tlink devhead+%TC%¥LIB¥c0s.obj+chgoem,
  chgoem, /c, %TC%¥LIB¥cs
```

(3) Turbo C++ 1.0

```
tasm /ML /DTurboCPP devhead;
tcc -c -w chgoem.c
tlink devhead+%TC%¥LIB¥c0s.obj+chgoem,
  chgoem, /c, %TC%¥LIB¥cs
```

(4) Borland C++ 3.00

```
tasm /ML /DBorlandCPP devhead;
bcc -c -w chgoem.c
tlink devhead+%TC%¥LIB¥c0s.obj+chgoem,
  chgoem, /c, %TC%¥LIB¥cs
```

なお，%TC%はTurbo C/C++，Borland C++のディレクトリを示す環境変数で，バッチファイルとして記述しています．

◆おわりに

筆者の場合，PC-9801シリーズをメインマシンとしてDOS/Vマシン，J-3100シリーズ，FM-TOWNSなど多くの機種を使用する機会があるため，機種判別は必須の課題です．これを極力簡単にしようとして考えたのがOEMのシリアル番号による対応です．

こうしておけば必要に応じて**リスト1**を改善していけば，各種のユーティリティは変更する必要がなくなります．いろいろなパソコンを使用する機会のある方は機種判別のひとつの実例として検討してみてください．

<参考文献>

1)中島信行，「DOS/Vコマンド&ユーティリティの全貌」，BootStrap Project-2 No.3
2)中島信行，「V-Text対応アプリケーションの作成」，BootStrap Project-2 No.5
3)中島信行，「MS-DOSポータブルプログラミング」，ソフトバンク
4)中島信行，「Cによるメモリ管理技法」，別冊インターフェース，CQ出版社
5)小高輝真，「98の未公開機能を探る!!」，The BASIC '90年11月号
6)矢沢宏行，「PC-9801シリーズ用MS-DOSの内部ワークエリアを公開する」，The BASIC '90年11月号
7)中野弾，「PC-9801シリーズのシステム管理領域を追う」，プロセッサ'90年10月号
8)小高輝真・清水和文・速水祐，「PC-9801スーパーテクニック」，アスキー出版局
9)「PC-9801シリーズテクニカルデータブック」，アスキー出版局
10)「PC-9801シリーズテクニカルデータブック増補版」，アスキー出版局
11)「PC-9801シリーズテクニカルデータブック増補改定版」，アスキー出版局
12)「PC-9801シリーズテクニカルデータブック増補改定2版」，アスキー出版局
13)「IBM Personal System/2 and Personal Computer BIOS Interface Technical Reference」，IBM
14) Ralf Brown and Jim Kyle，「PC INTERRUPTS」，Addison-Wesley Publishing Company, Inc.
15) Peter Norton & Richard Wilton，「THE IBM PC & PS/2プログラマーズガイド」，㈱翔泳社
16)「MS-DOS Version 5.0 Programmer's Reference」，Microsoft PRESS
17)「日本語MS-DOS V5.0プログラマーズリファレンスFM R-80/70/60/50，FM-TOWNS」，富士通㈱
18)「日本語MS-DOS V5.0アドバンストガイドFM R-80/70/60/50，FM-TOWNS」，富士通㈱
19)千葉憲昭，「FM-TOWNSテクニカルデータブック」，アスキー出版局
20)「富士通FMRシリーズ徹底解析マニュアル」，BNN
21)「PC-9800シリーズMS-DOS 5.0Aプログラマーズリファレンスマニュアル VOL.1」，NEC
22)「PC-9800シリーズMS-DOS 5.0Aプログラマーズリファレンスマニュアル VOL.2」，NEC
23)「PC-9800シリーズMS-DOS 5.0Aユーザーズリファレンスマニュアル」，NEC
24)「PC-9800シリーズMS-DOS 5.0Aプログラム開発ツールマニュアル」，NEC
25)「PC-9800シリーズMS-DOS 3.3Bプログラマーズリファレンスマニュアル Vol.1」，NEC
26)「PC-9800シリーズMS-DOS 3.3Bプログラマーズリファレンスマニュアル Vol.2」，NEC
27)「PC-9800シリーズMS-DOS 3.3Bユーザーズリファレンスマニュアル」，NEC
28)「PC-9800シリーズMS-DOS 3.3Bプログラム開発ツールマニュアル」，NEC
29)「PC-9800シリーズMS-DOS 3.1 プログラマーズリファレンスマニュアル」，NEC
30)「PC-9800シリーズMS-DOS 3.1 マクロアセンブラマニュアル」，NEC
31)「PC-9800シリーズMS-DOS 3.1 ユーザーズリファレンスマニュアル」，NEC
32)「IBM DOSバージョンJ5.0/V BIOSインターフェース技術解説書」，日本アイ・ビー・エム㈱
33)「IBM DOSバージョンJ5.0, J5.0/V技術解説書」，日本アイ・ビー・エム㈱
34)「IBM DOSバージョンJ5.0/Vユーザーズ・ガイド」，日本アイ・ビー・エム㈱
35)「IBM DOSバージョンJ4.0/V BIOSインターフェース技術解説書」，日本アイ・ビー・エム㈱
36)「IBM DOSバージョンJ4.0/V技術解説書」，日本アイ・ビー・エム㈱
37)「J-3100シリーズテクニカルマニュアル(ソフトウェア編)V3.1」，㈱東芝
38)「J-3100シリーズテクニカルマニュアル(ソフトウェア編)V2.11」，㈱東芝
39)「Microsoft C 6.00A Professional Development System Programming Guide」，Microsoft
40)「Microsoft C 6.00A Professional Development System Reference」，Microsoft
41)「Microsoft C/C++ 7.0A Environment and Utilitiles」，Microsoft
42)「Microsoft C/C++ 7.0A Index/Error Messages」，Microsoft
43)「Microsoft C/C++ 7.0A Class Libraries Reference」，Microsoft
44)「Microsoft C/C++ 7.0A Class Libraries User's Guide」，Microsoft
45)「Microsoft C/C++ 7.0A C Language Reference」，Microsoft
46)「Microsoft C/C++ 7.0A C++ Language Reference」，Microsoft
47)「Microsoft C/C++ 7.0A C++ Programming Guide」，Microsoft
48)「Microsoft C/C++ 7.0A C++ Tutorial」，Microsoft
49)「Microsoft C/C++ランタイムライブラリリファレンスVersion 7」，監修マイクロソフト㈱，発行㈱アスキー
50)「Microsoft Macro Assembler 5.10 Programmer's Guide」，Microsoft
51)「Microsoft Macro Assembler 6.0 Programming Guide」，Microsoft
52)「TURBO C 2.0 USER'S GUIDE」，BORLAND
53)「TURBO C 2.0 REFERENCE GUIDE」，BORLAND
54)「TURBO C++ 1.0 INTRODUCTION」，BORLAND
55)「TURBO C++ 1.0 USER'S GUIDE」，BORLAND
56)「TURBO C++ 1.0 PROGRAMMER'S GUIDE」，BORLAND
57)「TURBO C++ 1.0 LIBRARY REFERENCE Vol.1」，BORLAND
58)「TURBO C++ 1.0 LIBRARY REFERENCE Vol.2」，BORLAND
59)「BORLAND C++ 3.0 USER'S GUIDE」，BORLAND
60)「BORLAND C++ 3.0 PROGRAMMER'S GUIDE」，BORLAND
61)「BORLAND C++ 3.0 LIBRARY REFERENCE Vol.1」，BORLAND
62)「BORLAND C++ 3.0 LIBRARY REFERENCE Vol.2」，BORLAND
63)「BORLAND C++ 3.0 UTILITIES GUIDE」，BORLAND
64)「BORLAND C++ 3.0 QUICK REFERENCE GUIDE」，BORLAND
65)「TURBO ASSEMBLER 1.0 USER'S GUIDE」，BORLAND
66)「TURBO ASSEMBLER 1.0 REFERENCE GUIDE」，BORLAND
67)「TURBO ASSEMBLER 3.0 USER'S GUIDE」，BORLAND
68)「TURBO ASSEMBLER 3.0 QUICK REFERENCE GUIDE」，BORLAND

リスト1●CHGOEM.C

```
/***************************************************************************
*       MS-DOS OEM の ｼﾘｱﾙ番号 検査ﾕｰﾃｨﾘﾃｨ  ﾊﾞｰｼﾞｮﾝ 0.03    1993.10.17. *
*       Copyright (C)   N.Nakashima                  作成    1993.08.15. *
***************************************************************************
*       ﾌｧｲﾙ名 : chgoem.c                                                  *
***************************************************************************

    使用法 1. : device=chgoem.exe [-N<ｼﾘｱﾙ番号>]

    使用法 2. : chgoem [-N<ｼﾘｱﾙ番号>]

    <ｼﾘｱﾙ番号> が指定されると OEM の指定の ｼﾘｱﾙ番号に変更します.
    ｽｲｯﾁ -N だけで <ｼﾘｱﾙ番号> を省略すると次のように変更します.

                       変更値 ｵﾘｼﾞﾅﾙ値
        PC-9800 ｼﾘｰｽﾞ    FFh     ←
        IBM DOS/V        00h     ←
        J-3100 ｼﾘｰｽﾞ     29h     ←
        AX ﾏｼﾝ           16h     色々
        IBM PC           69h     色々
        FM-TOWNS         D4h     FFh
        MS-DOS 汎用      4Dh     色々
        不明             55h     色々

    ﾊﾞｰｼﾞｮﾝ 0.01    1993.08.15.
    ﾊﾞｰｼﾞｮﾝ 0.02    1993.08.22.    EXE型の ﾃﾞﾊﾞｲｽ･ﾄﾞﾗｲﾊﾞ に変更
    ﾊﾞｰｼﾞｮﾝ 0.03    1993.10.17.    PC DOS J6.1/V Alpha 版に対応

Microsoft (R) C Compiler  Version 6.00A / 7.0A

masm /ML devhead;
cl /Ot /J /Gs /W3 /Zp /Fechgoem devhead.obj chgoem.c /link /st:0x2800 /cp:0x500
cl /Ot /J /Gs /W3 /Zpl /Fechgoem devhead.obj chgoem.c /link slibcr+libh /st:0x2800
 /cp:0x500

Turbo C  Version 2.0

tasm /ML devhead;
tcc -c -w chgoem.c
tlink devhead+%TC%¥LIB¥c0s.obj+chgoem,chgoem,/c,%TC%¥LIB¥cs

Turbo C++  Version 1.0

tasm /ML /DTurboCPP devhead;
tcc -c -w chgoem.c
tlink devhead+%TC%¥LIB¥c0s.obj+chgoem,chgoem,/c,%TC%¥LIB¥cs

Borland C++  Version 3.00

tasm /ML /DBorlandCPP devhead;
bcc -c -w chgoem.c
tlink devhead+%TC%¥LIB¥c0s.obj+chgoem,chgoem,/c,%TC%¥LIB¥cs

*/

#include <stdio.h>
#include <ctype.h>
#include <string.h>
#include <stdlib.h>
#include <conio.h>
#include <dos.h>

#ifndef MK_FP
#define MK_FP(s,o) ((void far *)(((unsigned long)(s) << 16) | (unsigned)(o)))
#endif

#if     defined( __BORLANDC__ )     /* Borland C++                    */
#elif   defined( __TURBOC__ )       /* Turbo C/C++                    */
#define inpw        inport

int far _fmemcmp( char far *p1, char far *p2, size_t n );
#elif   defined( LSI_C )            /* LSI C                          */
#include <farstr.h>
#include <machine.h>

#define _fmemcmp    far_memcmp
#elif   defined( _MSC_VER )         /* MS-C 6.00A, 7.0A               */
#else                               /* MS-C 5.10                      */
int far _fmemcmp( char far *p1, char far *p2, size_t n );
#endif

#define FALSE           0           /* 偽   */
#define TRUE            ( !FALSE )  /* 真   */

struct devhead_t {                          /* ｷｬﾗｸﾀ･ﾃﾞﾊﾞｲｽ ﾃﾞﾊﾞｲｽ･ﾍｯﾀﾞ      */
    struct devhead_t far *pnextdev;         /* 次の ﾃﾞﾊﾞｲｽ･ﾍｯﾀﾞ を指すﾎﾟｲﾝﾀ */
    unsigned int  uattr;                    /* 属性                          */
    void ( near *strategy )();              /* ｽﾄﾗﾃｼﾞ･ｴﾝﾄﾘ･ﾎﾟｲﾝﾀ の ﾎﾟｲﾝﾀ   */
    void ( near *int_entry )();             /* 割込ｴﾝﾄﾘ･ﾎﾟｲﾝﾀ の ﾎﾟｲﾝﾀ      */
    unsigned char devname[ 8 ];             /* ｷｬﾗｸﾀ･ﾃﾞﾊﾞｲｽ ﾌｧｲﾙ名          */
};

struct reqhead_t {                          /* ﾘｸｴｽﾄ･ﾍｯﾀﾞの定義             */
    unsigned char cmdlen;                   /* ｺﾏﾝﾄﾞ長                       */
    unsigned char unitcode;                 /* ﾕﾆｯﾄ番号                      */
    unsigned char cmdcode;                  /* ｺﾏﾝﾄﾞ番号                     */
    unsigned      status;                   /* ｽﾃｰﾀｽ                         */
    unsigned char dmy[ 8 ];
    unsigned char unitnum;                  /* ｻﾎﾟｰﾄ する ﾕﾆｯﾄ数             */
    void far *pbrkadr;                      /* ﾌﾘｰ･ﾒﾓﾘ 開始ｱﾄﾞﾚｽ             */
    unsigned char far *poption;             /* ｵﾌﾟｼｮﾝ･ｽｲｯﾁ                   */
};

typedef struct {                            /* ﾊﾟｿｺﾝ機種情報                 */
    unsigned type;                          /* ﾊﾟｿｺﾝ機種                     */
    unsigned videomode;                     /* ﾋﾞﾃﾞｵ･ﾓｰﾄﾞ                    */
    unsigned rows;                   /* 一画面の行数 (ｽｸﾛｰﾙする範囲の行数)   */
    unsigned cols;                          /* 一行当りの桁数                */
    unsigned total_rows;                    /* 画面全体の行数                */
    unsigned fjapan;                        /* 日本語ﾓｰﾄﾞ                    */
    void far *ptvramadr;                    /* ﾃｷｽﾄ VRAM ｱﾄﾞﾚｽ               */
```

```
} pctype_t;

extern struct devhead_t far sdevhead;   /* ﾃﾞﾊﾞｲｽ･ﾍｯﾀﾞ                     */
void init( void ), done( void ), cmderr( void );
int chgoem( void ), getpctype( pctype_t * );
unsigned atoh( char far * );

struct reqhead_t far *preqhead = NULL;  /* ﾘｸｴｽﾄ･ﾍｯﾀﾞ ｱﾄﾞﾚｽ                */
pctype_t pctype = { 0 };                /* ﾊﾟｿｺﾝ機種情報                   */
int fsetoemno = FALSE;                  /* OEM の ｼﾘｱﾙ番号 変更ﾌﾗｸﾞ        */
int oemno = -1;                         /* OEM の ｼﾘｱﾙ番号                 */

void inthdl( void )                     /* 割込ﾙｰﾁﾝ                        */
{
    static void ( *functbl[] )( void ) = {
        init,                           /*  0  初期設定                    */
        done,                           /*  1  媒体検査                    */
        done,                           /*  2  BPB 作成                    */
        cmderr,                         /*  3  IOCTL 入力                  */
        done,                           /*  4  入力 ( 入力待ち )           */
        done,                           /*  5  連続非破壊読み込み          */
        done,                           /*  6  入力ｽﾃｰﾀｽ                   */
        done,                           /*  7  入力ﾌﾗｯｼｭ                   */
        done,                           /*  8  出力                        */
        done,                           /*  9  出力 & ﾍﾞﾘﾌｧｲ               */
        done,                           /* 10  出力ｽﾃｰﾀｽ                   */
        done,                           /* 11  出力ﾊﾞｯﾌｧ ﾌﾗｯｼｭ             */
        cmderr                          /* 12  IOCTL 出力                  */
    };

    if ( preqhead->cmdcode < sizeof( functbl ) / sizeof( functbl[ 0 ] ) )
        functbl[ preqhead->cmdcode ](); /* 各ｺﾏﾝﾄﾞ 実行                    */
    else
        cmderr();                       /* ｺﾏﾝﾄﾞ ｴﾗｰ                       */
}

void done( void )
{
    preqhead->status = 0x0100;          /* DONE ﾋﾞｯﾄ の ｾｯﾄ                */
}

void cmderr( void )                     /* ｺﾏﾝﾄﾞ ｴﾗｰ                       */
{
    preqhead->status = 0x8103;
                              /* ｴﾗｰ･ﾋﾞｯﾄ, DONE ﾋﾞｯﾄ ｾｯﾄ : 無効なｺﾏﾝﾄﾞ    */
}

void init( void )                       /* 初期化処理                      */
{
    char far *popt = ( char far * )preqhead->poption;   /* ｵﾌﾟｼｮﾝ･ｽｲｯﾁ   */

    while ( *popt > ' ' )
        popt++;                         /* ﾌｧｲﾙ名を ｽｷｯﾌﾟ                  */
    while ( *popt != '¥n' && *popt != '¥r' ) {
        if ( *popt == '-' || *popt == '/' ) {
            ++popt;
            switch ( toupper( *popt ) ) {      /* ｽｲｯﾁ         ﾁｪｯｸ        */
            case 'N' :
                if ( isxdigit( popt[ 1 ] ) ) {
                    oemno = ( unsigned char )atoh( ++popt );
                                        /* OEM の ｼﾘｱﾙ番号                 */
                    while ( isxdigit( *popt ) )
                        popt++;
                    popt--;
                } else
                    oemno = -1;
                fsetoemno = TRUE;       /* OEM の ｼﾘｱﾙ番号 変更            */
                break;
            default :
                break;
            }
        }
        popt++;
    }

    {
        int version = bdos( 0x30, 0, 0x00 );/* MS-DOS ﾊﾞｰｼﾞｮﾝ番号の取得 */

        _osmajor = ( unsigned char )version;
        _osminor = ( unsigned char )( version >> 8 );
    }
    getpctype( &pctype );               /* ﾊﾟｿｺﾝ種別の取得                 */
    chgoem();                           /* OEM の ｼﾘｱﾙ番号 検査            */

    /* ﾃﾞﾊﾞｲｽ･ﾄﾞﾗｲﾊﾞ の組込みを取り止め  */

    sdevhead.uattr &= 0x7fff;           /* ﾌﾞﾛｯｸ･ﾃﾞﾊﾞｲｽ に変更             */
    preqhead->unitnum = 0;              /* ﾕﾆｯﾄ数 = 0                      */
#if     defined( LSI_C )
    {
        struct SREGS segs;

        segread( &segs );
        preqhead->pbrkadr = MK_FP( segs.cs, &sdevhead );
                                        /* ﾌﾘｰ･ﾒﾓﾘ 開始ｱﾄﾞﾚｽ                */
    }
#else
    preqhead->pbrkadr = ( void far * )&sdevhead;/* ﾌﾘｰ･ﾒﾓﾘ 開始ｱﾄﾞﾚｽ      */
                                /* LSI C-86 ﾊﾞｸﾞ : CS ではなく DS ｾｯﾄ      */
#endif
    preqhead->status  = 0x0100;         /* DONE ﾋﾞｯﾄ の ｾｯﾄ                */
}

#define NEW98OEM        0xff            /* PC-9800 ｼﾘｰｽﾞ                   */
#define NEWdosvOEM      0x00            /* IBM DOS/V                       */
#define NEW31OEM        0x29            /* J-3100 ｼﾘｰｽﾞ                    */
#define NEWaxOEM        0x16            /* AX ﾏｼﾝ                          */
#define NEWibmOEM       'i'             /* IBM PC                          */
#define NEWtownsOEM ( 0x80 + 'T' )      /* FM-TOWNS                        */
#define NEWmsOEM        'M'             /* MS-DOS 汎用                     */
#define NEWunknownOEM   'U'             /* 不明                            */

/*          ﾊﾟｿｺﾝ機種          */

```

```
#define pctypeDOSVj     'V'          /* DOS/V        日本語ﾓｰﾄﾞ          */
#define pctypeDOSVe     'v'          /* DOS/V        英語ﾓｰﾄﾞ            */
#define pctypeAXj       'A'          /* AX ﾏｼﾝ       日本語ﾓｰﾄﾞ          */
#define pctypeAXe       'a'          /* AX ﾏｼﾝ       英語ﾓｰﾄﾞ            */
#define pctypeJ3100j    'J'          /* J-3100 ｼﾘｰｽﾞ 日本語ﾓｰﾄﾞ          */
#define pctypeJ3100e    'j'          /* J-3100 ｼﾘｰｽﾞ 英語ﾓｰﾄﾞ            */
#define pctypeIBMe      'e'          /* IBM PC 英語 DOS または英語ﾓｰﾄﾞ    */
#define pctypePC9801    0x98         /* PC-9801 ｼﾘｰｽﾞ                    */
#define pctypeTOWNS ( 0x80 | 'T' )   /* FM-TOWNS                         */
#define pctypeMSDOSj    'M'          /* MS-DOS 汎用 日本語ﾓｰﾄﾞ           */
#define pctypeMSDOSe    'm'          /* MS-DOS 汎用 英語ﾓｰﾄﾞ             */
#define pctypeOTHER     0x00         /* 上記以外の機種                   */
#define videoPC9801     0xff         /* PC-9801 ｼﾘｰｽﾞ 仮の ﾋﾞﾃﾞｵ･ﾓｰﾄﾞ    */
#define videoTOWNS      0xfe         /* FM-TOWNS      仮の ﾋﾞﾃﾞｵ･ﾓｰﾄﾞ    */

typedef struct {                          /* MCB 構造体                  */
    char     mark;                 /* ﾒﾓﾘ管理用の ﾏｰｸ ( 'M' or 'Z' )     */
    unsigned owner;                       /* OWNER ID                    */
    unsigned mcbsiz;                      /* ﾒﾓﾘ･ﾌﾞﾛｯｸ の ﾊﾟﾗｸﾞﾗﾌ数       */
    char     mcbdmy[ 3 ];
    char     progname[ 8 ];        /* Owner ﾌｧｲﾙ名 ( DOS 4.x 以降 )      */
} mcb_t;

void getargs( int , char ** ), usage( void ),
     utonh( unsigned, char *, int ), dos_cputs( char * );
char far *msdoscmp( unsigned, unsigned, char *, size_t );
char far *getoemsetadr( void );
char far *getoemgetadr( unsigned, unsigned, unsigned * );
void far *gettwndos( void );
void far *getdosvar( void );
int getoem( void ), isdoshma( void );

char dos6a[] = { 0x89, 0x04, 0x89, 0x4c, 0x04, 0x89, 0x5c, 0x02 };
char dos5a[] = { 0x89, 0x04, 0x89, 0x5c, 0x02, 0x89, 0x4c, 0x04 };
char dos4a[] = { 0x89, 0x5c, 0x02, 0x89, 0x4c, 0x04 };
char dos3a[] = { 0x89, 0x5c, 0x02, 0x89, 0x4c, 0x04, 0x89, 0x04 };

int main( int argc, char *argv[] )
{
    getpctype( &pctype );                 /* ﾊﾟｿｺﾝ種別の取得             */
    getargs( argc, argv );                /* 引数の取得                  */
    return( chgoem() );                   /* OEM の ｼﾘｱﾙ番号 検査        */
}

int chgoem( void )                        /* OEM の ｼﾘｱﾙ番号 検査        */
{
    int oldoem;                           /* 元の OEM の ｼﾘｱﾙ番号        */
    int newoem;                           /* 変更する OEM の ｼﾘｱﾙ番号    */
    char far *pdosvar;                    /* 内部変数領域の ｱﾄﾞﾚｽ        */
    char far *pfar1, far *pfar2, far *pfar;
    unsigned bhoffset;                    /* BH ﾚｼﾞｽﾀ への ｵﾌｾｯﾄ          */
    char msgbuf[ 10 ];

    switch ( pctype.type ) {              /* ﾊﾟｿｺﾝ機種                   */
    case pctypePC9801 :                   /* PC-9801 ｼﾘｰｽﾞ               */
        newoem = NEW98OEM;                /* 変更する OEM の ｼﾘｱﾙ番号    */
        dos_cputs( "PC-9801 ｼﾘｰｽﾞ です." );
        break;
    case pctypeDOSVj :                    /* DOS/V        日本語ﾓｰﾄﾞ     */
        newoem = NEWdosvOEM;              /* 変更する OEM の ｼﾘｱﾙ番号    */
        dos_cputs( "DOS/V 日本語ﾓｰﾄﾞ です." );
        break;
    case pctypeDOSVe :                    /* DOS/V        英語ﾓｰﾄﾞ       */
        newoem = NEWdosvOEM;              /* 変更する OEM の ｼﾘｱﾙ番号    */
        dos_cputs( "DOS/V US mode." );
        break;
    case pctypeJ3100j :                   /* J-3100 ｼﾘｰｽﾞ 日本語ﾓｰﾄﾞ     */
        newoem = NEW31OEM;                /* 変更する OEM の ｼﾘｱﾙ番号    */
        dos_cputs( "J-3100 ｼﾘｰｽﾞ 日本語ﾓｰﾄﾞ です." );
        break;
    case pctypeJ3100e :                   /* J-3100 ｼﾘｰｽﾞ 英語ﾓｰﾄﾞ       */
        newoem = NEW31OEM;                /* 変更する OEM の ｼﾘｱﾙ番号    */
        dos_cputs( "J-3100 US mode." );
        break;
    case pctypeAXj :                      /* AX ﾏｼﾝ       日本語ﾓｰﾄﾞ     */
        newoem = NEWaxOEM;                /* 変更する OEM の ｼﾘｱﾙ番号    */
        dos_cputs( "AX ﾏｼﾝ 日本語ﾓｰﾄﾞ です." );
        break;
    case pctypeAXe :                      /* AX ﾏｼﾝ       英語ﾓｰﾄﾞ       */
        newoem = NEWaxOEM;                /* 変更する OEM の ｼﾘｱﾙ番号    */
        dos_cputs( "AX ﾏｼﾝ US mode." );
        break;
    case pctypeIBMe :               /* IBM PC 英語 DOS または英語ﾓｰﾄﾞ    */
        newoem = NEWibmOEM;               /* 変更する OEM の ｼﾘｱﾙ番号    */
        dos_cputs( "IBM PC Compatible." );
        break;
    case pctypeTOWNS :                    /* FM-TOWNS                    */
        newoem = NEWtownsOEM;             /* 変更する OEM の ｼﾘｱﾙ番号    */
        dos_cputs( "FM-TOWNS です." );
        break;
    case pctypeMSDOSj :                   /* MS-DOS 汎用 日本語ﾓｰﾄﾞ      */
        newoem = NEWmsOEM;                /* 変更する OEM の ｼﾘｱﾙ番号    */
        dos_cputs( "MS-DOS 汎用 日本語ﾓｰﾄﾞ です." );
        break;
    case pctypeMSDOSe :                   /* MS-DOS 汎用 英語ﾓｰﾄﾞ        */
        newoem = NEWmsOEM;                /* 変更する OEM の ｼﾘｱﾙ番号    */
        dos_cputs( "MS-DOS US mode." );
        break;
    case pctypeOTHER :                    /* 上記以外                    */
    default :
        newoem = NEWunknownOEM;           /* 変更する OEM の ｼﾘｱﾙ番号    */
        dos_cputs( "Unknown PC." );
        break;
    }
    oldoem = getoem();                    /* 元の OEM の ｼﾘｱﾙ番号        */
    dos_cputs( "  OEM Serial No. : " );
    utonh( oldoem, msgbuf, 2 );
    dos_cputs( msgbuf );

    if ( oemno < 0 )
        oemno = newoem;
    if ( !fsetoemno || oemno == oldoem ) {
        dos_cputs( "¥n" );
        return( 0 );                      /* OEM の ｼﾘｱﾙ番号の未変更     */
```

```c
    }
    if ( ( pfar1 = getoemsetadr() ) == NULL
      || ( pfar2 = getoemgetadr( FP_OFF( pfar1 ) - 0x20,
                                 FP_OFF( pfar1 ), &bhoffset ) ) == NULL
      || ( pfar1 <= pfar2 ) ) {
        dos_cputs( "¥nUnknown MS-DOS Version.¥n" );
        return( 1 );
    }
    pdosvar = getdosvar();              /* 内部変数領域の ｱﾄﾞﾚｽ の取得  */
    pfar = MK_FP( FP_SEG( pdosvar ), *( unsigned far * )pfar2 + bhoffset );
    *pfar = ( char )oemno;
    dos_cputs( pctype.fjapan ? " → " : " -> " );
    utonh( getoem(), msgbuf, 2 );
    dos_cputs( msgbuf );
    dos_cputs( "¥n" );
    return( 0 );
}

/*
    MSDOS.SYS ﾌﾟﾛｸﾞﾗﾑ領域の検索
*/
char far *msdoscmp( unsigned stadr,      /* 検索開始ｵﾌｾｯﾄ･ｱﾄﾞﾚｽ          */
                    unsigned endadr,     /* 検索終了ｵﾌｾｯﾄ･ｱﾄﾞﾚｽ          */
                    char     *pcode,     /* 検索 ﾃﾞｰﾀ                   */
                    register size_t ulen ) /* 検索 ﾊﾞｲﾄ数                */
{
    char far *pmsdos;                    /* MSDOS.SYS ﾌﾟﾛｸﾞﾗﾑ領域        */

    if ( pctype.type == pctypeTOWNS && _osmajor == 5 ) {/* FM-TOWNS DOS5*/
        pmsdos = isdoshma() ? ( char far * )0xffff0000L : gettwndos();
    } else
        pmsdos = isdoshma() ? ( char far * )0xffff0000L : getdosvar();
    if ( stadr )
        pmsdos = MK_FP( FP_SEG( pmsdos ), stadr );
    endadr -= ulen;
    do {
        if ( !_fmemcmp( pmsdos, pcode, ulen ) )
            return( pmsdos );
    } while ( ( unsigned )( unsigned long )++pmsdos <= endadr );
    return( ( char far * )NULL );
}

char far *getoemsetadr( void )       /* OEM の ｼﾘｱﾙ番号の ｾｯﾄ箇所の取得  */
{
    char far *pfar;

    switch ( _osmajor ) {                /* MS-DOS ﾊﾞｰｼﾞｮﾝ番号           */
    case 6 :
        pfar = msdoscmp( 0, 0xffff, dos6a, sizeof( dos6a ) );
        break;
    case 5 :
        pfar = msdoscmp( 0, 0xffff, dos5a, sizeof( dos5a ) );
        break;
    case 4 :
        pfar = msdoscmp( 0, 0xffff, dos4a, sizeof( dos4a ) );
        break;
    case 3 :
        pfar = msdoscmp( 0, 0xffff, dos3a, sizeof( dos3a ) );
        break;
    }
    if ( pfar != NULL )
        return( pfar );
    if ( ( pfar = msdoscmp( 0, 0xffff, dos5a, sizeof( dos5a ) ) ) != NULL )
        return( pfar );
    if ( ( pfar = msdoscmp( 0, 0xffff, dos6a, sizeof( dos6a ) ) ) != NULL )
        return( pfar );
    if ( ( pfar = msdoscmp( 0, 0xffff, dos3a, sizeof( dos3a ) ) ) != NULL )
        return( pfar );
    return( msdoscmp( 0, 0xffff, dos4a, sizeof( dos4a ) ) );
}

/*
    OEM の ｼﾘｱﾙ番号の ﾘｰﾄﾞ箇所の取得
*/
char far *getoemgetadr( unsigned stadr,      /* 検索開始ｵﾌｾｯﾄ･ｱﾄﾞﾚｽ      */
                        unsigned endadr,     /* 検索終了ｵﾌｾｯﾄ･ｱﾄﾞﾚｽ      */
                        unsigned *poffset ) /* BH ﾚｼﾞｽﾀ への ｵﾌｾｯﾄ      */
{
    char far *pfar;

    switch ( _osmajor ) {                    /* MS-DOS ﾊﾞｰｼﾞｮﾝ番号       */
    case 6 :
        pfar = msdoscmp( stadr, endadr, "¥xc5¥x0e", 2 );
                                             /* lds cx,[????] 検索       */
        if ( pfar != NULL
          && *( unsigned far * )( pfar + 4 ) == 0xdb8c ) {/* mov bx,ds ?*/
            *poffset = 3;                    /* BH ﾚｼﾞｽﾀ への ｵﾌｾｯﾄ      */
            return( pfar + 2 );
        }                                    /* no break;    */
    case 5 :
    case 4 :
    case 3 :
        pfar = msdoscmp( stadr, endadr, "¥x8b¥x1e", 2 );
                                             /* mov bx,[????] 検索       */
        if ( pfar != NULL ) {
            *poffset = 1;                    /* BH ﾚｼﾞｽﾀ への ｵﾌｾｯﾄ      */
            return( pfar + 2 );
        }
        break;
    }
    return( NULL );
}

/*
    FM-TOWNS MS-DOS 5.0 の DOS=LOW の場合の MS-DOS.SYS ｺｰﾄﾞ領域の取得
*/
void far *gettwndos( void )
{
    unsigned far *pdosvar;                   /* 内部変数領域の ｱﾄﾞﾚｽ      */
    unsigned int2e_seg = *( unsigned far * )( 0x2eL * 4 + 2 );
    mcb_t far *pmcb;

    pdosvar = getdosvar();                   /* 内部変数領域の ｱﾄﾞﾚｽ      */
    for ( pmcb = MK_FP( *( pdosvar - 1 ), 0 ); FP_SEG( pmcb ) < int2e_seg; ) {
```

```
        if ( pmcb->owner == 8           /* OWNER ID : ｼｽﾃﾑ ?              */
          && *( int far * )pmcb->progname == 0x4353 ) {
                                        /* Owner ﾌｧｲﾙ名 : "SC"             */
            return( MK_FP( FP_SEG( pmcb ) + 1, 0 ) );
        }
        if ( pmcb->mark == 'Z' )
            break;
        if ( pmcb->mark != 'M' ) {
            return( NULL );
        }
        *( ( unsigned far * )&pmcb + 1 ) += pmcb->mcbsiz + 1;
    }
    return( NULL );
}

void getargs( int argc, char *argv[] )  /* 引数の取得                      */
{
    while ( --argc ) {
        char *p = *++argv;

        if ( *p == '-' || *p == '/' ) {         /* ｽｲｯﾁ指定     ﾁｪｯｸ     */
            ++p;
            switch ( toupper( *p ) ) {          /* ｽｲｯﾁ         ﾁｪｯｸ     */
            case 'N' :
                if ( isxdigit( p[ 1 ] ) ) {
                    oemno = ( unsigned char )atoh( p + 1 );
                                        /* OEM の ｼﾘｱﾙ番号                 */
                } else
                    oemno = -1;
                fsetoemno = TRUE;       /* OEM の ｼﾘｱﾙ番号 変更            */
                break;
            default :
                usage();                /* 使用法表示                      */
                break;
            }
        } else
            usage();                    /* 使用法表示                      */
    }
}

void usage( void )
{
    static char *jmsg[] = {
        "MS-DOS OEM の ｼﾘｱﾙ番号 検査ﾕｰﾃｨﾘﾃｨ  ﾊﾞｰｼﾞｮﾝ 0.03¥n",
        "Copyright (C) 1993     N.Nakashima¥n¥n",
        "使用法 1. : device=chgoem.exe [-N<ｼﾘｱﾙ番号>]¥n¥n",
        "使用法 2. : chgoem [-N<ｼﾘｱﾙ番号>]¥n¥n",
        "<ｼﾘｱﾙ番号> が指定されると OEM の指定の ｼﾘｱﾙ番号に変更します.¥n",
        "ｽｲｯﾁ -N だけで <ｼﾘｱﾙ番号> を省略すると次のように変更します.¥n¥n",
        "                  変更値 ｵﾘｼﾞﾅﾙ値¥n",
        "    PC-9800 ｼﾘｰｽﾞ   FFh     ←¥n",
        "    IBM DOS/V       00h     ←¥n",
        "    J-3100 ｼﾘｰｽﾞ    29h     ←¥n",
        "    AX ﾏｼﾝ          16h     色々¥n",
        "    IBM PC          69h     色々¥n",
        "    FM-TOWNS        D4h     FFh¥n",
        "    MS-DOS 汎用     4Dh     色々¥n",
        "    不明            55h     色々¥n",
        NULL
    };
    static char *emsg[] = {
        "MS-DOS OEM Serial No. Checker  Version 0.03¥n",
        "Copyright (C) 1993  N.Nakashima¥n¥n",
        "Usage : chgoem [-N<Serial No.>]¥n¥n",
        NULL
    };
    register char **pmsg = pctype.fjapan ? jmsg : emsg;

    while ( *pmsg )
        dos_cputs( *pmsg++ );
    exit( 1 );
}

/*
    16 進 文字列 → Bin. 2 ﾊﾞｲﾄ 変換
*/
unsigned atoh( char far *str /* 16 進文字列 格納ｱﾄﾞﾚｽ */ )
{
    register unsigned val;              /* Bin. 変換値                     */
    register unsigned ch;

    while ( isspace( *str ) )
        ++str;                          /* ｽﾍﾟｰｽ を ｽｷｯﾌﾟ                 */
    val = 0;
    while ( ( ch = *str++ ) != '¥0' ) {
        if ( !isxdigit( ch ) )
            break;                      /* 16進文字列 ﾁｪｯｸ                 */
        ch = toupper( ch );
        val = ( val << 4 ) + ( ( ch < 'A' ) ? ( ch - '0' ) : ( ch - 'A' + 10 ) );
    }
    return( val );
}

/*
    符号無整数 → 16 進文字列
*/
void utonh( unsigned val,               /* 符号無整数                      */
            register char *str,         /* 16 進文字列 格納ｱﾄﾞﾚｽ           */
            register int  n )           /* 桁数                            */
{
    unsigned v;                         /* 符号無整数                      */

    str += n;                           /* ｽﾄﾘﾝｸﾞ 最後のｱﾄﾞﾚｽ+1 ｾｯﾄ        */
    *str-- = '¥0';
    do {
        v     = val;
        val  >>= 4;
        v     -= ( val << 4 );          /* 余り                            */
        *str-- = ( char )( ( ( v < 10 ) ? ( v + '0' ) : ( v - 10 + 'A' ) ) );
                                        /* 余り Ascii 変換                 */
    } while ( --n > 0 && val > 0 );
    while ( --n >= 0 )
        *str-- = '0';                   /* '0' ｾｯﾄ                         */
}
```

```
void dos_cputs( char *str )
{
    while ( *str ) {
        if ( *str == '¥n' )
            bdos( 0x06, '¥r', 0x00 );
        bdos( 0x06, *str++, 0x00 );
    }
}

int getoem( void )                      /* MS-DOS OEM の ｼﾘｱﾙ番号 取得  */
{
    union REGS regs;

    regs.x.ax = 0x3000;
    intdos( &regs, &regs );
    return( regs.h.bh & 0xff );         /* MS-DOS OEM の ｼﾘｱﾙ番号       */
}

/*
    返り値  内部変数領域の ｱﾄﾞﾚｽ
*/
void far *getdosvar( void )             /* 内部変数領域の ｱﾄﾞﾚｽ の取得  */
{
    struct SREGS segs;
    union REGS regs;

    regs.h.ah = 0x52;
    intdosx( &regs, &regs, &segs );
    return( MK_FP( segs.es, regs.x.bx ) );
}

int isdoshma( void )                    /* MS-DOS が HMA にあるかどうか */
{
    union REGS regs;

    regs.x.ax = 0x3306;                 /* Get MS-DOS Version           */
    intdos( &regs, &regs );
    return( ( regs.h.al != 0xff ) && ( regs.h.dh & 0x10 ) != 0 );
}

int isibmpc( void ), isdosv( void ), isax( void ), isj3100( void ),
    isDBCSmode( void ), ispc9801( void ), istowns( void );
void far *dosv_getvramadr( void );
void dosv_getvmode( pctype_t *ptype );

/*
    ﾊﾟｿｺﾝ機種の判定

    返り値 ﾊﾟｿｺﾝ機種     = pctypeDOSVj     : DOS/V          日本語ﾓｰﾄﾞ
                         = pctypeDOSVe     : DOS/V          英語ﾓｰﾄﾞ
                         = pctypeAXj       : AX ﾏｼﾝ         日本語ﾓｰﾄﾞ
                         = pctypeAXe       : AX ﾏｼﾝ         英語ﾓｰﾄﾞ
                         = pctypeJ3100j    : J-3100 ｼﾘｰｽﾞ   日本語ﾓｰﾄﾞ
                         = pctypeJ3100e    : J-3100 ｼﾘｰｽﾞ   英語ﾓｰﾄﾞ
                         = pctypeIBMe      : IBM PC 英語 DOS または英語ﾓｰﾄﾞ
                         = pctypePC9801    : PC-9801 ｼﾘｰｽﾞ
                         = pctypeTOWNS     : FM-TOWNS
                         = pctypeOTHER     : 上記以外の機種
*/
int getpctype( pctype_t *ptype )        /* ﾊﾟｿｺﾝ種別の取得               */
{
    ptype->type = pctypeOTHER;          /* ﾊﾟｿｺﾝ機種 : 不明              */
/*  getenvPCTYPE( ptype ); */           /* 環境変数 PCTYPE の判定        */
    if ( ptype->type == pctypeDOSVj     /* DOS/V         日本語ﾓｰﾄﾞ      */
      || ptype->type == pctypeDOSVe     /* DOS/V         英語ﾓｰﾄﾞ        */
      || ptype->type == pctypeAXj       /* AX ﾏｼﾝ        日本語ﾓｰﾄﾞ      */
      || ptype->type == pctypeAXe       /* AX ﾏｼﾝ        英語ﾓｰﾄﾞ        */
      || ptype->type == pctypeJ3100j    /* J-3100 ｼﾘｰｽﾞ  日本語ﾓｰﾄﾞ      */
      || ptype->type == pctypeJ3100e    /* J-3100 ｼﾘｰｽﾞ  英語ﾓｰﾄﾞ        */
      || ptype->type == pctypeIBMe  /* IBM PC 英語 DOS または英語ﾓｰﾄﾞ    */
      || ( ptype->type == pctypeOTHER && isibmpc() ) ) {
        int type;                       /* IBM PC 互換機                 */

        dosv_getvmode( ptype );         /* ﾋﾞﾃﾞｵ情報の取得               */
        if ( ptype->type == pctypeDOSVj /* DOS/V         日本語ﾓｰﾄﾞ      */
          || ptype->type == pctypeDOSVe /* DOS/V         英語ﾓｰﾄﾞ        */
          || isdosv() ) {               /* DOS/V ﾏｼﾝ ?                   */
            type = isDBCSmode();        /* 日本語ﾓｰﾄﾞ と英語ﾓｰﾄﾞ の判定 */
            ptype->fjapan = ( type == pctypeDOSVj );    /* 日本語ﾓｰﾄﾞ   */
        } else if ( ptype->type == pctypeAXj    /* AX ﾏｼﾝ  日本語ﾓｰﾄﾞ   */
                 || ptype->type == pctypeAXe    /* AX ﾏｼﾝ  英語ﾓｰﾄﾞ     */
                 || isax() != pctypeOTHER ) {
            type = isax();              /* AX ﾏｼﾝ                        */
            ptype->fjapan = ( type == pctypeAXj );      /* 日本語ﾓｰﾄﾞ   */
        } else if ( ptype->type == pctypeJ3100j /* J-3100 ｼﾘｰｽﾞ 日本語ﾓｰﾄﾞ */
                 || ptype->type == pctypeJ3100e /* J-3100 ｼﾘｰｽﾞ 英語ﾓｰﾄﾞ */
                 || isj3100() != pctypeOTHER ) {
            type = isj3100();           /* J-3100 ｼﾘｰｽﾞ                  */
            ptype->fjapan = ( type == pctypeJ3100j );   /* 日本語ﾓｰﾄﾞ   */
            ptype->ptvramadr = NULL;    /* ﾃｷｽﾄ VRAM ｱﾄﾞﾚｽ               */
        } else {
            type = pctypeIBMe;      /* IBM PC 英語 DOS または英語ﾓｰﾄﾞ    */
        }
        return( ptype->type = type );   /* ﾊﾟｿｺﾝ機種                     */
    } else if ( ptype->type == pctypePC9801 /* PC-9801 ｼﾘｰｽﾞ             */
             || ( ptype->type == pctypeOTHER && ispc9801() ) ) {
                                        /* PC-9801 ｼﾘｰｽﾞ                 */
        static unsigned char far *pfuncdisp = ( unsigned char far * )0x600111L;
                                        /* = 0 : ﾌｧﾝｸｼｮﾝ･ｷｰ未表示        */
                                        /* = 1 : ﾌｧﾝｸｼｮﾝ･ｷｰ表示          */
        static unsigned char far *pscrollin = ( unsigned char far * )0x600112L;
                                        /* ｽｸﾛｰﾙ行数    ﾌｧﾝｸｼｮﾝ･ｷｰ内容   */
                                        /*              表示    未表示   */
                                        /* 20 行ﾓｰﾄﾞ    18      19       */
                                        /* 25 行ﾓｰﾄﾞ    23      24       */

        ptype->type = pctypePC9801;     /* ﾊﾟｿｺﾝ機種 : PC9801            */
        ptype->videomode = videoPC9801; /* ﾋﾞﾃﾞｵ･ﾓｰﾄﾞ                    */
        ptype->rows = *pscrollin + 1;
                              /* 一画面の行数 (ｽｸﾛｰﾙする範囲の行数)      */
        ptype->cols = 80;               /* 一行当りの桁数                */
        if ( *pfuncdisp == 0 ) {        /* ﾌｧﾝｸｼｮﾝ･ｷｰ未表示 ?            */
            ptype->total_rows = ptype->rows;        /* 画面全体の行数    */
```

```
        } else {                           /* 最下行に ﾌｧﾝｸｼｮﾝ･ｷｰ を表示   */
            ptype->total_rows = ptype->rows + 1;    /* 画面全体の行数   */
        }
        ptype->fjapan    = TRUE;          /* 日本語ﾓｰﾄﾞ                   */
        ptype->ptvramadr = ( void far * )0xa0000000L;/* ﾃｷｽﾄ VRAM ｱﾄﾞﾚｽ */
        return( pctypePC9801 );
    } else if ( ptype->type == pctypeTOWNS  /* FM-TOWNS                 */
            || ( ptype->type == pctypeOTHER && istowns() ) ) {
                                           /* FM-TOWNS                  */
        ptype->type = pctypeTOWNS;         /* ﾊﾟｿｺﾝ機種 : FM-TOWNS      */
        ptype->videomode = videoTOWNS;  /* ﾋﾞﾃﾞｵ･ﾓｰﾄﾞ                   */
        ptype->rows = 24;         /* 一画面の行数 (ｽｸﾛｰﾙする範囲の行数)  */
        ptype->cols = 80;                  /* 一行当りの桁数            */
        ptype->total_rows = 25;            /* 画面全体の行数            */
        ptype->fjapan    = TRUE;           /* 日本語ﾓｰﾄﾞ                */
        ptype->ptvramadr = NULL;           /* ﾃｷｽﾄ VRAM ｱﾄﾞﾚｽ           */
        return( pctypeTOWNS );
    }
    ptype->videomode = 0xff;               /* ﾋﾞﾃﾞｵ･ﾓｰﾄﾞ                */
    ptype->rows = 24;             /* 一画面の行数 (ｽｸﾛｰﾙする範囲の行数)  */
    ptype->cols = 80;                      /* 一行当りの桁数            */
    ptype->total_rows = 25;                /* 画面全体の行数            */
/*  ptype->fjapan    = FALSE;  */          /* 英語ﾓｰﾄﾞ                  */
    ptype->ptvramadr = NULL;               /* ﾃｷｽﾄ VRAM ｱﾄﾞﾚｽ           */
    return( ptype->type );
}

int isibmpc( void )                        /* IBM PC 互換機の判定       */
{
#define KEYTOP  0x001e                     /* ｷｰﾎﾞｰﾄﾞ･ﾊﾞｯﾌｧ先頭ｱﾄﾞﾚｽ    */
#define KEYEND  ( 0x001e + 16 * 2 )        /* ｷｰﾎﾞｰﾄﾞ･ﾊﾞｯﾌｧ最終ｱﾄﾞﾚｽ    */
    static unsigned far *prs232_base = ( unsigned far * )0x400000L;
                                           /* RS-232C の ﾎﾟｰﾄ･ｱﾄﾞﾚｽ     */
    static unsigned far *printer_base = ( unsigned far * )0x400008L;
                                           /* ﾌﾟﾘﾝﾀ  の ﾎﾟｰﾄ･ｱﾄﾞﾚｽ      */
    static unsigned far *pkeytop = ( unsigned far * )0x400080L;
                                           /* ｷｰﾎﾞｰﾄﾞ･ﾊﾞｯﾌｧ先頭ﾎﾟｲﾝﾀ    */
    static unsigned far *pkeyend = ( unsigned far * )0x400082L;
                                           /* ｷｰﾎﾞｰﾄﾞ･ﾊﾞｯﾌｧ末尾ﾎﾟｲﾝﾀ    */

    if ( *pkeytop == KEYTOP && *pkeyend == KEYEND )
        return( TRUE );                    /* ｷｰﾎﾞｰﾄﾞ･ﾊﾞｯﾌｧ ﾎﾟｲﾝﾀ ﾁｪｯｸ */
    if ( ( *prs232_base == 0x03f8 || *prs232_base == 0x02f8 )
                                           /* RS-232C の ﾎﾟｰﾄ･ｱﾄﾞﾚｽ ﾁｪｯｸ */
      && ( *printer_base == 0x0378 || *printer_base == 0x0278
        || *printer_base == 0x03bc ) )     /* ﾌﾟﾘﾝﾀ  の ﾎﾟｰﾄ･ｱﾄﾞﾚｽ ﾁｪｯｸ */
        return( TRUE );
    return( FALSE );
}

int isdosv( void )/* DOS/V ﾏｼﾝ の判定 ( IBM PC 互換機である必要がある. ) */
{
#if     defined( _MSC_VER )
    unsigned pseg = 0, poff = 0;

    _asm push   bp
    _asm mov    ax,5000h             /* ﾌｫﾝﾄの読み取り機能のｱﾄﾞﾚｽの取得  */
    _asm mov    bx,0100h                   /* 2 ﾊﾞｲﾄ文字ｾｯﾄ                */
    _asm mov    dx,1010h                   /* 16 x 16 ﾄﾞｯﾄ 全角文字ﾊﾟﾀｰﾝ   */
    _asm mov    bp,0000h                   /* ｺｰﾄﾞ･ﾍﾟｰｼﾞの省略値           */
    _asm mov    es,bp                      /* 返り値 es = 0000h            */
    _asm int    15h
    _asm pop    bp
    _asm jc     isdosve
    _asm cmp    ah,00h
    _asm jnz    isdosve
    _asm mov    pseg,es
    _asm mov    poff,bx                    /* es:bx ﾌｫﾝﾄの読み取り機能のｱﾄﾞﾚｽ */
isdosve:
    return( pseg != 0 || poff != 0x0100 );
#elif   defined( __TURBOC__ )
    struct REGPACK regs;

    regs.r_ax = 0x5000;                    /* ﾌｫﾝﾄの読み取り機能のｱﾄﾞﾚｽの取得 */
    regs.r_bx = 0x0100;                    /* 2 ﾊﾞｲﾄ文字ｾｯﾄ                */
    regs.r_dx = 0x1010;                    /* 16 x 16 ﾄﾞｯﾄ 全角文字ﾊﾟﾀｰﾝ   */
    regs.r_bp = 0x0000;                    /* ｺｰﾄﾞ･ﾍﾟｰｼﾞの省略値           */
    regs.r_es = 0x0000;                    /* 返り値 es = 0000             */
    intr( 0x15, &regs );
    return( !( regs.r_flags & 0x0001 )  /* ｷｬﾘｰ == 0 ?                   */
          && !( regs.r_ax & 0xff00 )      /* ah   == 0 ?                   */
          && ( regs.r_es != 0x0000 || regs.r_bx != 0x0100 ) );
#elif   defined( __LSIC__ )
    struct SREGS segs;
    union REGS regs;

    regs.x.ax = 0x5000;                    /* ﾌｫﾝﾄの読み取り機能のｱﾄﾞﾚｽの取得 */
    regs.x.bx = 0x0100;                    /* 2 ﾊﾞｲﾄ文字ｾｯﾄ                */
    regs.x.dx = 0x1010;                    /* 16 x 16 ﾄﾞｯﾄ 全角文字ﾊﾟﾀｰﾝ   */
    regs.x.bp = 0x0000;                    /* ｺｰﾄﾞ･ﾍﾟｰｼﾞの省略値           */
    segs.es    = 0x0000;                   /* 返り値 es = 0000             */
    int86x( 0x15, &regs, &regs, &segs );
    return( !( regs.x.flags & 0x0001 )  /* ｷｬﾘｰ == 0 ?                   */
          && !( regs.x.ax & 0xff00 )      /* ah   == 0 ?                   */
          && ( segs.es != 0x0000 || regs.x.bx != 0x0100 ) );
#else                                      /* MS-C 5.1 : 未ｻﾎﾟｰﾄ          */
    union REGS regs;
    struct SREGS segs;

    segs.es = regs.x.di = 0;
    regs.h.ah = 0xfe;                      /* 疑似ﾋﾞﾃﾞｵ･ﾊﾞｯﾌｧ ｱﾄﾞﾚｽ の取得 */
    int86x( 0x10, &regs, &regs, &segs );
    return( segs.es != 0 || regs.x.di != 0 );
#endif
}

/*
    AX ﾏｼﾝ の判定 ( IBM PC 互換機である必要がある. )

    返り値  pctypeAXj   : AX ﾏｼﾝ 日本語ﾓｰﾄﾞ
            pctypeAXe   : AX ﾏｼﾝ 英語ﾓｰﾄﾞ
            pctypeOTHER : AX ﾏｼﾝ 以外
*/
int isax( void )
```

```
{
    union REGS regs;

    regs.x.ax = 0x5001;                  /* 日本語ﾓｰﾄﾞと英語ﾓｰﾄﾞの取得   */
    int86( 0x10, &regs, &regs );
    if ( regs.h.al == 00 ) {
        if ( regs.x.bx == 1 )            /* 英語ﾓｰﾄﾞ ?                    */
            return( pctypeAXe );         /* AX ﾏｼﾝ 英語ﾓｰﾄﾞ               */
        else if ( regs.x.bx == 81 )
            return( pctypeAXj );         /* AX ﾏｼﾝ 日本語ﾓｰﾄﾞ             */
    }
    return( pctypeOTHER );               /* AX ﾏｼﾝ 以外                   */
}

/*
    J-3100 ｼﾘｰｽﾞ の判定 ( IBM PC 互換機である必要がある. )

    返り値  pctypeJ3100j : J-3100 ｼﾘｰｽﾞ 日本語ﾓｰﾄﾞ
            pctypeJ3100e : J-3100 ｼﾘｰｽﾞ 英語ﾓｰﾄﾞ
            pctypeOTHER  : J-3100 ｼﾘｰｽﾞ 以外
*/
int isj3100( void )
{
    union REGS regs;
    struct SREGS segs;

    regs.h.ah = 0x88;                    /* 外字ﾌｫﾝﾄ･ﾃｰﾌﾞﾙ･ｱﾄﾞﾚｽ の取得 */
    segs.es = regs.x.bx = 0;             /* es:bx = 0:0                   */
    int86x( 0x10, &regs, &regs, &segs );
    if ( segs.es != 0 || regs.x.bx != 0 ) {
        static unsigned char far *pvideomode = ( unsigned char far * )0x400049L;
        unsigned char videomode = ( unsigned char )( *pvideomode & 0x7f );
                                         /* ﾋﾞﾃﾞｵ･ﾓｰﾄﾞ番号                */

        if ( ( 0x64 <= videomode && videomode <= 0x67 )
          || ( 0x74 <= videomode && videomode <= 0x77 )
          || videomode == 0x62 || videomode == 0x72 )
            return( pctypeJ3100j );      /* J-3100 ｼﾘｰｽﾞ 日本語ﾓｰﾄﾞ       */
        else
            return( pctypeJ3100e );      /* J-3100 ｼﾘｰｽﾞ 英語ﾓｰﾄﾞ         */
    }
    return( pctypeOTHER );               /* J-3100 ｼﾘｰｽﾞ 以外             */
}

/*
    日本語ﾓｰﾄﾞ と英語ﾓｰﾄﾞ の判定

    返り値  pctypeDOSVj : 日本語ﾓｰﾄﾞ
            pctypeDOSVe : 英語ﾓｰﾄﾞ
            pctypeIBMe  : 不明
*/
int isDBCSmode( void )
{
    union REGS regs;
    struct SREGS segs;

    regs.x.ax = 0x6300;                  /* DBCS ﾍﾞｸﾀ情報の取得           */
    segs.ds = regs.x.si = 0;             /* ds:si = 0:0                   */
    intdosx( &regs, &regs, &segs );
    if ( segs.ds != 0 || regs.x.si != 0 ) {
        if ( *( unsigned far * )MK_FP( segs.ds, regs.x.si ) ) {
            /* 日本語以外の 2 ﾊﾞｲﾄ文字ｾｯﾄ を ｻﾎﾟｰﾄ する場合は
               81-9F, E0-FC を確認する必要があるが、ここでは確認しない. */
            return( pctypeDOSVj );       /* 日本語ﾓｰﾄﾞ                    */
        } else
            return( pctypeDOSVe );       /* 英語ﾓｰﾄﾞ                      */
    }
    return( pctypeIBMe );                /* DOS ﾌｧﾝｸｼｮﾝ 6300h 未ｻﾎﾟｰﾄ     */
}

int ispc9801( void )                     /* PC-9801 ｼﾘｰｽﾞ の判定          */
{
#define KEY98TOP    0x0502               /* ｷｰﾎﾞｰﾄﾞ･ﾊﾞｯﾌｧ                */
#define KEY98END    ( 0x0502 + 16 * 2 ) /* ｷｰﾎﾞｰﾄﾞ･ﾊﾞｯﾌｧ                */
    static unsigned far *pkey98head = ( unsigned far * )0x0524L;
                                         /* ｷｰﾎﾞｰﾄﾞ･ﾊﾞｯﾌｧ次読出ｱﾄﾞﾚｽ      */
    static unsigned far *pkey98tail = ( unsigned far * )0x0526L;
                                         /* ｷｰﾎﾞｰﾄﾞ･ﾊﾞｯﾌｧ次格納ｱﾄﾞﾚｽ      */
    static unsigned char far *pkey98cnt = ( unsigned char far * )0x0528L;
                                         /* ｷｰﾎﾞｰﾄﾞ･ﾊﾞｯﾌｧ格納ｶｳﾝﾄ数       */
    static unsigned char far *pfuncdisp = ( unsigned char far * )0x600111L;
                                         /* = 0 : ﾌｧﾝｸｼｮﾝ･ｷｰ未表示        */
                                         /* = 1 : ﾌｧﾝｸｼｮﾝ･ｷｰ表示          */
    static unsigned char far *pscrollin = ( unsigned char far * )0x600112L;
                                         /* ｽｸﾛｰﾙ行数    ﾌｧﾝｸｼｮﾝ･ｷｰ内容 */
                                         /*              表示    未表示 */
                                         /* 20 行ﾓｰﾄﾞ    18      19     */
                                         /* 25 行ﾓｰﾄﾞ    23      24     */
    static unsigned char far *pline25   = ( unsigned char far * )0x600113L;
                                         /* 25 行ﾓｰﾄﾞ                     */
                                         /* = 0 : 20 行ﾓｰﾄﾞ               */
                                         /* = 1 : 25 行ﾓｰﾄﾞ               */

    /*  ｷｰﾎﾞｰﾄﾞ･ﾊﾞｯﾌｧ ﾎﾟｲﾝﾀ ﾁｪｯｸ    */

    if ( ( KEY98TOP <= *pkey98head && *pkey98head < KEY98END )
                                         /* ｷｰﾎﾞｰﾄﾞ･ﾊﾞｯﾌｧ次読出ｱﾄﾞﾚｽ ﾁｪｯｸ  */
      && ( KEY98TOP <= *pkey98tail && *pkey98tail < KEY98END ) ) {
                                         /* ｷｰﾎﾞｰﾄﾞ･ﾊﾞｯﾌｧ次格納ｱﾄﾞﾚｽ ﾁｪｯｸ  */
        unsigned keyend;                 /* ｷｰﾎﾞｰﾄﾞ･ﾊﾞｯﾌｧ次格納ｱﾄﾞﾚｽ      */

        if ( ( keyend = *pkey98head + *pkey98cnt * 2 ) >= KEY98END )
            keyend -= KEY98END - KEY98TOP;
        if ( keyend == *pkey98tail )/* ｷｰﾎﾞｰﾄﾞ･ﾊﾞｯﾌｧ次格納ｱﾄﾞﾚｽ ﾁｪｯｸ  */
            return( TRUE );
    }

    /*  行数ﾁｪｯｸ    */

    if ( *pfuncdisp < 3 ) {
        if ( *pline25 == 0 ) {           /* 20 行ﾓｰﾄﾞ ?                   */
            if ( *pfuncdisp == 0 ) {     /* ﾌｧﾝｸｼｮﾝ･ｷｰ未表示 ?             */
                return( *pscrollin == 19 );
                             /* ﾌｧﾝｸｼｮﾝ･ｷｰ未表示時の ｽｸﾛｰﾙ行数 ﾁｪｯｸ  */
```

```
            } else {
                return( *pscrollin == 18 );
                                    /* ﾌｧﾝｸｼｮﾝ･ｷｰ表示時の ｽｸﾛｰﾙ行数 ﾁｪｯｸ     */
            }
        } else if ( *pline25 == 1 ) {   /* 25 行ﾓｰﾄﾞ ?                    */
            if ( *pfuncdisp == 0 ) {    /* ﾌｧﾝｸｼｮﾝ･ｷｰ未表示 ?              */
                return( *pscrollin == 24 );
                                    /* ﾌｧﾝｸｼｮﾝ･ｷｰ未表示時の ｽｸﾛｰﾙ行数 ﾁｪｯｸ  */
            } else {
                return( *pscrollin == 23 );
                                    /* ﾌｧﾝｸｼｮﾝ･ｷｰ表示時の ｽｸﾛｰﾙ行数 ﾁｪｯｸ     */
            }
        }
    }
    return( FALSE );
}

int istowns( void )                         /* FM-TOWNS の判定               */
{
    int data;

    data = inpw( 0x30 );                    /* CPU 識別ﾚｼﾞｽﾀ                 */
    return( ( inp( 0x20e ) & 0xfe ) == 0    /* ﾄﾞﾗｲﾌﾞ･ｽｲｯﾁ･ﾚｼﾞｽﾀ             */
        && ( 1 <= ( unsigned char )data && ( unsigned char )data <= 7 )
        && ( data & 0xff00 ) != 0xff00 );
}

void dosv_getvmode( pctype_t *ptype )       /* ﾋﾞﾃﾞｵ情報の取得               */
{
#if     defined( _MSC_VER )
    _asm push bp
    _asm mov ah,0x0f                        /* ﾋﾞﾃﾞｵ情報の取得               */
    _asm int 0x10
    _asm push ax
    _asm mov ax,0x1130                      /* ﾌｫﾝﾄ･ﾃｰﾌﾞﾙ情報の取得          */
    _asm mov bh,1           /* 0:010Ch ( INT 43h ) の ｴﾝﾄﾘ･ｱﾄﾞﾚｽ を返す */
    _asm int 0x10               /* 返り値として bp ﾚｼﾞｽﾀ が破壊される    */
    _asm pop ax
    _asm pop bp
    _asm mov di,ptype                       /* ﾋﾞﾃﾞｵ情報構造体への ﾎﾟｲﾝﾀ     */
    _asm mov bl,ah                          /* 一行当りの桁数                */
    _asm mov bh,0
    _asm mov [di]ptype.cols,bx              /* 一行当りの桁数                */
    _asm and ax,7fh
    _asm mov [di]ptype.videomode,ax         /* ﾋﾞﾃﾞｵ･ﾓｰﾄﾞ                    */
    _asm mov dh,0
    _asm inc dx
    _asm mov [di]ptype.rows,dx  /* 一画面の行数 (ｽｸﾛｰﾙする範囲の行数)    */
    _asm mov ax,0x1d02              /* ｷｰﾎﾞｰﾄﾞ･ｼﾌﾄ標識域の ｽﾃｰﾀｽ の取得 */
    _asm sub bx,bx
    _asm int 0x10
    _asm add dx,bx
    _asm mov [di]ptype.total_rows,dx
                                /* 画面全体の行数 ( bx = 標識域の行数 ) */
#elif   defined( __TURBOC__ )
    struct REGPACK regs;

    regs.r_ax = 0x0f00;                     /* ﾋﾞﾃﾞｵ情報の取得               */
    intr( 0x10, &regs );
    ptype->videomode = regs.r_ax & 0x7f;/* ﾋﾞﾃﾞｵ･ﾓｰﾄﾞ                     */
    ptype->cols = regs.r_ax >> 8;           /* 一行当りの桁数                */
    regs.r_ax = 0x1130;                     /* ﾌｫﾝﾄ･ﾃｰﾌﾞﾙ情報の取得          */
    regs.r_bx = 0x0100;     /* 0:010Ch ( INT 43h ) の ｴﾝﾄﾘ･ｱﾄﾞﾚｽ を返す */
    intr( 0x10, &regs );
    ptype->rows = ( unsigned char )( regs.r_dx + 1 );
                                /* 一画面の行数 (ｽｸﾛｰﾙする範囲の行数)    */
    regs.r_ax = 0x1d02;             /* ｷｰﾎﾞｰﾄﾞ･ｼﾌﾄ標識域の ｽﾃｰﾀｽ の取得 */
    regs.r_bx = 0;
    intr( 0x10, &regs );
    ptype->total_rows = ptype->rows + regs.r_bx;
                                /* 画面全体の行数 ( bx = 標識域の行数 ) */
#elif   defined( LSI_C )
    union REGS regs;

    regs.h.ah = 0x0f;                       /* ﾋﾞﾃﾞｵ情報の取得               */
    int86( 0x10, &regs, &regs );
    ptype->videomode = regs.h.al & 0x7f;/* ﾋﾞﾃﾞｵ･ﾓｰﾄﾞ                     */
    ptype->cols = regs.h.ah;                /* 一行当りの桁数                */
    regs.x.ax = 0x1130;                     /* ﾌｫﾝﾄ･ﾃｰﾌﾞﾙ情報の取得          */
    regs.h.bh = 1;          /* 0:010Ch ( INT 43h ) の ｴﾝﾄﾘ･ｱﾄﾞﾚｽ を返す */
    int86( 0x10, &regs, &regs );/* 返り値として bp ﾚｼﾞｽﾀ が破壊される    */
    ptype->rows = regs.h.dl + 1;/* 一画面の行数 (ｽｸﾛｰﾙする範囲の行数)    */
    regs.x.ax = 0x1d02;             /* ｷｰﾎﾞｰﾄﾞ･ｼﾌﾄ標識域の ｽﾃｰﾀｽ の取得 */
    regs.x.bx = 0;
    int86( 0x10, &regs, &regs );
    ptype->total_rows = ptype->rows + regs.x.bx;
                                /* 画面全体の行数 ( bx = 標識域の行数 ) */
#endif
    ptype->ptvramadr = dosv_getvramadr();   /* ﾃｷｽﾄ VRAM ｱﾄﾞﾚｽ              */
}

void far *dosv_getvramadr( void )       /* 疑似ﾋﾞﾃﾞｵ･ﾊﾞｯﾌｧ ｱﾄﾞﾚｽ の取得 */
{
    union REGS regs;
    struct SREGS segs;
    int videomode;                          /* ﾋﾞﾃﾞｵ･ﾓｰﾄﾞ                    */

    regs.h.ah = 0x0f;                       /* ﾋﾞﾃﾞｵ情報の取得               */
    int86( 0x10, &regs, &regs );
    videomode = regs.h.al & 0x7f;           /* ﾋﾞﾃﾞｵ･ﾓｰﾄﾞ                    */
    if ( videomode == 0x03 ) {              /* ﾋﾞﾃﾞｵ･ﾓｰﾄﾞ == 3 ?             */
        segs.es = 0xb800;
        regs.x.di = 0;                      /* es:di = b800:0000             */
    } else if ( videomode == 0x70 ) {       /* ﾋﾞﾃﾞｵ･ﾓｰﾄﾞ == 70h ?           */
        segs.es = regs.x.di = 0;            /* es:di = 0000:0000             */
    } else
        return( NULL );
    regs.h.ah = 0xfe;                   /* 疑似ﾋﾞﾃﾞｵ･ﾊﾞｯﾌｧ ｱﾄﾞﾚｽ の取得 */
    int86x( 0x10, &regs, &regs, &segs );
    return( MK_FP( segs.es, regs.x.di ) );
}

#if !defined( _MSC_VER ) && !defined( __BORLANDC__ ) && !defined( LSI_C )
int far _fmemcmp( char far *p1, char far *p2, size_t n )
```

```
{
    while ( n-- ) {
        int c;

        if ( ( c = *p1++ - *p2++ ) != 0 )
            return( c );
    }
    return( 0 );
}
#endif
```

リスト2●DEVHEAD.ASM

```
        name    DEVHEAD
        ;
        page    66,132
        ;
        title   C 言語による ﾃﾞﾊﾞｲｽ･ﾄﾞﾗｲﾊﾞ ﾍｯﾀﾞ･ﾌｧｲﾙ    ﾊﾞｰｼﾞｮﾝ 0.01
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
;       C 言語による ﾃﾞﾊﾞｲｽ･ﾄﾞﾗｲﾊﾞ ﾍｯﾀﾞ･ﾌｧｲﾙ  ﾊﾞｰｼﾞｮﾝ 0.01  1992.11.15. ;
;       Copyright (C)   N.Nakashima                  作成  1992.11.15. ;
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
;       ﾌｧｲﾙ名 : devhead.asm                                            ;
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
;
;       ﾃﾞﾊﾞｲｽ･ﾄﾞﾗｲﾊﾞ 作成時には、本ﾌﾟﾛｸﾞﾗﾑ を一番最初に ﾘﾝｸ してください.
;
;       Microsoft (R) C Compiler  Version 5.10 / 6.00A / 7.0A
;
;       masm /ML devhead;
;
;       Turbo C  Version 2.0
;
;       tasm /ML devhead;
;
;       Turbo C++  Version 1.0
;
;       tasm /ML /DTurboCPP devhead;
;
;       Borland C++  Version 3.00
;
;       tasm /ML /DBorlandCPP devhead;
;
;       ﾊﾞｰｼﾞｮﾝ 0.01    1992.11.15.
;

        public  strategy,interrupt,_sdevhead,_devatr
        extrn   _preqhead:dword,_inthdl:near

DEVHEAD         segment word public 'CODE'
DEVHEAD         ends
_TEXT           segment word public 'CODE'
_TEXT           ends
ifdef   TurboCPP
_FARDATA        segment para public 'FAR_DATA'
_FARDATA        ends
_OVERLAY_       segment para public 'OVRINFO'
_OVERLAY_       ends
_INIT_          segment word public 'INITDATA'
_INIT_          ends
_INITEND_       segment byte public 'INITDATA'
_INITEND_       ends
_EXIT_          segment word public 'EXITDATA'
_EXIT_          ends
_EXITEND_       segment byte public 'EXITDATA'
_EXITEND_       ends
_1STUB_         segment para public 'STUBSEG'
_1STUB_         ends
endif
ifdef   BorlandCPP
_FARDATA        segment para public 'FAR_DATA'
_FARDATA        ends
_FARBSS         segment para public 'FAR_BSS'
_FARBSS         ends
_OVERLAY_       segment para public 'OVRINFO'
_OVERLAY_       ends
_1STUB_         segment para public 'STUBSEG'
_1STUB_         ends
endif
_DATA           segment para public 'DATA'
_DATA           ends
ifdef   BorlandCPP
_CVTSEG         segment word public 'DATA'
_CVTSEG         ends
_SCNSEG         segment word public 'DATA'
_SCNSEG         ends
_CONST          segment word public 'CONST'
_CONST          ends
_INIT_          segment word public 'INITDATA'
_INIT_          ends
_INITEND_       segment byte public 'INITDATA'
_INITEND_       ends
_EXIT_          segment word public 'EXITDATA'
_EXIT_          ends
_EXITEND_       segment byte public 'EXITDATA'
_EXITEND_       ends
endif
_BSS            segment word public 'BSS'
        dw      512 dup (?)     ; ﾛｰｶﾙ･ｽﾀｯｸ領域
stktop  label   word
_BSS            ends
ifdef   BorlandCPP
_BSSEND         segment byte public 'BSS'
_BSSEND         ends
_STACK          segment stack 'STACK'
_STACK          ends
endif

ifdef   BorlandCPP
DGROUP  group   _DATA,_CVTSEG,_SCNSEG,_CONST,_BSS,_BSSEND
IGROUP  group   _INIT_,_INITEND_
EGROUP  group   _EXIT_,_EXITEND_
```

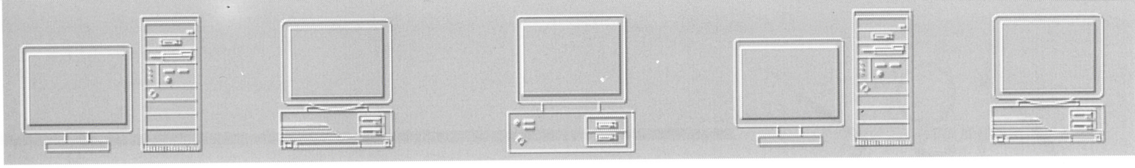
```
else
DGROUP  group   _DATA,_BSS
ifdef   TurboCPP
IGROUP  group   _INIT_,_INITEND_
EGROUP  group   _EXIT_,_EXITEND_
endif
endif
DEVHEAD segment word public 'CODE'
        assume  cs:DEVHEAD, ds:DGROUP, ss:DGROUP
;
;       デバイス・ヘッダ
;
_sdevhead dd    -1                      ; 次の デバイス・ヘッダ への ポインタ
_devatr dw      08000h                  ; デバイス属性
        dw      strategy                ; ストラテジ エントリー・ポイント
        dw      interrupt               ; 割込  エントリー・ポイント
        db      '$DEVHEAD'              ; デバイス名
;
;       ストラテジ・ルーチン
;
dsreg   dw      DGROUP
strategy proc   far
        push    ds
        mov     ds,cs:[dsreg]
        mov     word ptr [DGROUP:_preqhead],bx  ; リクエスト・ヘッダ アドレス退避
        mov     word ptr [DGROUP:_preqhead+2],es
        pop     ds
        ret
strategy endp
;
;       割込みルーチン
;
interrupt proc  far
        push    ax
        push    bx
        push    cx
        push    dx
        push    bp
        push    si
        push    di
        push    ds
        push    es
        mov     ax,ss
        mov     es,ax
        mov     bp,sp
        mov     ax,DGROUP
        mov     ds,ax                   ; ds = DGROUP
        mov     ss,ax                   ; ss = DGROUP
        mov     sp,offset DGROUP:stktop ; ローカル・スタック 設定
        push    es                      ; 旧 SS レジスタ 退避
        push    bp                      ; 旧 SP レジスタ 退避
        cld
        call    far ptr inttext
        pop     bp
        pop     ss                      ; SS レジスタ 復帰
        mov     sp,bp                   ; SP レジスタ 復帰
        pop     es
        pop     ds
        pop     di
        pop     si
        pop     bp
        pop     dx
        pop     cx
        pop     bx
        pop     ax
        ret
interrupt endp
DEVHEAD ends
_TEXT   segment word public 'CODE'
        assume  cs:_TEXT
inttext proc    far
        call    _inthdl         ; near コール
        ret
inttext endp
_TEXT   ends
        end
```

リスト3●DOSファンクション30H部分のコード(PC9801シリーズ, MS-DOS5.0A)

```
FFFF:70B0 16          PUSH  SS
FFFF:70B1 1F          POP   DS
FFFF:70B2 8B1EB403    MOV   BX,[03B4] ; BH = OEMのシリアル番号
FFFF:70B6 8B0EB203    MOV   CX,[03B2] ; BL:CX = 24ビットのユーザー・シリアル番号
FFFF:70BA 3C01        CMP   AL,01
FFFF:70BC 7502        JNZ   70C0
FFFF:70BE 32FF        XOR   BH,BH     ; バージョン・フラグ
FFFF:70C0 1E          PUSH  DS
FFFF:70C1 8E1E3003    MOV   DS,[0330]
FFFF:70C5 A14000      MOV   AX,[0040] ; バージョン番号
FFFF:70C8 1F          POP   DS
FFFF:70C9 E82EF8      CALL  68FA
FFFF:70CC 8904        MOV   [SI],AX
FFFF:70CE 895C02      MOV   [SI+02],BX
FFFF:70D1 894C04      MOV   [SI+04],CX
FFFF:70D4 C3          RET
```

リスト4●DOSファンクション30H部分のコード(EPSON MS-DOS4.0)

```
0919:1DC5 16          PUSH  SS
0919:1DC6 1F          POP   DS
0919:1DC7 8B1EB403    MOV   BX,[03B4] ; BH = OEMのシリアル番号
0919:1DCB 8B0EB203    MOV   CX,[03B2] ; BL:CX = 24ビットのユーザー・シリアル番号
0919:1DCF A1120D      MOV   AX,[0D12] ; バージョン番号
0919:1DD2 E84DFB      CALL  1922
```



```
0919:1DD5 895C02        MOV   [SI+02],BX
0919:1DD8 894C04        MOV   [SI+04],CX
0919:1DDB 2E803E6111FF  CMP   Byte Ptr CS:[1161],FF
0919:1DE1 740D          JZ    1DF0
0919:1DE3 2E803E611100  CMP   Byte Ptr CS:[1161],00
0919:1DE9 7411          JZ    1DFC
0919:1DEB 2EFE0E6111    DEC   Byte Ptr CS:[1161]
0919:1DF0 2E833E5F1100  CMP   Word Ptr CS:[115F],+00
0919:1DF6 7404          JZ    1DFC
0919:1DF8 2EA15F11      MOV   AX,CS:[115F]
0919:1DFC 8904          MOV   [SI],AX
0919:1DFE C3            RET
```

リスト5●DOSファンクション30H部分のコード（PC-9801シリーズ MS-DOS 3.3B）

```
09F5:1691 16            PUSH  SS
09F5:1692 1F            POP   DS
09F5:1693 8B1E5403      MOV   BX,[0354] ; BH = OEMのシリアル番号
09F5:1697 8B0E5203      MOV   CX,[0352] ; BL:CX = 24ビットのユーザー・シリアル番号
09F5:169B A1780C        MOV   AX,[0C78] ; バージョン番号
09F5:169E E828FC        CALL  12C9
09F5:16A1 895C02        MOV   [SI+02],BX
09F5:16A4 894C04        MOV   [SI+04],CX
09F5:16A7 8904          MOV   [SI],AX
09F5:16A9 C3            RET
```

リスト6●DOSファンクション30H部分のコード（PC-DOS J6 1/V Alpha版）

```
FFFF:2793 36C50EB203    LDS   CX,SS:[03B2] ; BL:CX = 24ビットのユーザー・シリアル番号
FFFF:2798 8CDB          MOV   BX,DS        ; BH = OEMのシリアル番号
FFFF:279A 16            PUSH  SS
FFFF:279B 1F            POP   DS
FFFF:279C 3C01          CMP   AL,01
FFFF:279E 7502          JNZ   27A2
FFFF:27A0 32FF          XOR   BH,BH        ; バージョン・フラグ
FFFF:27A2 1E            PUSH  DS
FFFF:27A3 8E1E3003      MOV   DS,[0330]
FFFF:27A7 A14000        MOV   AX,[0040] ; バージョン番号
FFFF:27AA 1F            POP   DS
FFFF:27AB E88AF8        CALL  2038
FFFF:27AE 8904          MOV   [SI],AX
FFFF:27B0 894C04        MOV   [SI+04],CX
FFFF:27B3 895C02        MOV   [SI+02],BX
FFFF:27B6 C3            RET
```

TB TIME

Micro Device

MicroDevice

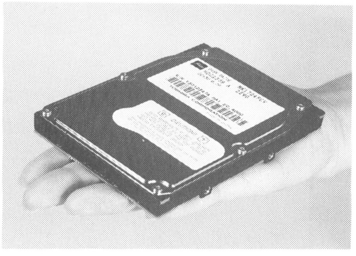
NEC，64ビットRISC搭載マルチチップモジュール

NEC（03-3798-6511）は，64ビットRISC型マイクロプロセッサ「VR4400」とキャッシュメモリを一つのパッケージに搭載したマルチチップモジュール「MR-4401-75」のサンプル出荷を開始した．同製品は，75MHz動作の「VR4400MC」と合計1MBのキャッシュメモリを179ピンPGAパッケージに組み込むことで，単体の「VR4400MC」を利用したシステムと比べ約3割の高速化を実現．「VR4000/4400PC」と上位互換のピン配列などの特徴を持つ．サンプル価格は30万円/個．

MicroDevice

日本AMD，PCI用SCSIコントローラ

日本AMD㈱（03-3346-7560）は，PCI対応のSCSIコントローラ「Am53C974（PCSCSI）」を発表した．同製品はSCSI-2（FAST SCSI）対応で，バスマスタ型DMA，PCIバスインターフェースからなる．またサポートソフトウェアはMS-DOSやOS/2などの主要なOSに対応している．サンプル価格は3,150円（千個/ロット）．



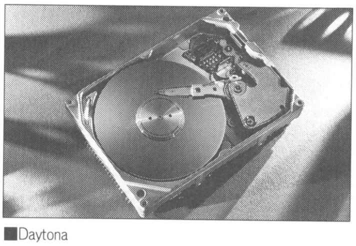
■東芝MK1724FCV

MicroDevice

Quantum，サブノート用ハードディスク「Daytona」など

日本クアンタム㈱（03-5952-6300）は，Quantum社が2.5インチHDDの新シリーズ「Quantum Daytona」シリーズを開発したと発表した．同シリーズには，サブノート型コンピュータ向け厚さ12.5mm，256MBならびに127MB，ノートブック型コンピュータ向け厚さ19mm，514MB，341MBの容量を持つモデルが含まれる．同シリーズは高レベルの耐衝撃性，低消費電力設計などの特徴を持つ．量産開始は1994年1月の予定．

また，同社はハードディスクドライブ用の高速PRML（Partial Response Maximum Likekihood）リードチャネルを開発．このPRMLを搭載した最初の製品として3.5インチドライブEmpire 1440（1.44GB），同2160（2.16GB）を発売した．両製品とも64MB/secのディスクバッファ間転送速度と5MB/secの持続転送速度を実現．SCSI-3インターフェイスに対応し，Fast/Wide SCSIをサポートしている．OEM向け「Empire 1440，2160」の量産開始は1994年1-3月期の予定．



■Daytona

MicroDevice

東芝，2.5インチ262MBハードディスク

㈱東芝（03-5484-6174）は，2.5インチ，厚さ1/2インチで262MBのハードディスク「MK1724FCV」を開発，サンプル出荷を開始した．同製品は平均シークタイム13ms，4,000rpm/secで，128KBのバッファメモリを内蔵している．インターフェースにはPC/ATを採用．サンプル価格は12万円で，量産開始は1994年1月から．

MicroDevice

富士通，MRヘッドを搭載2.5，3.5インチHDD

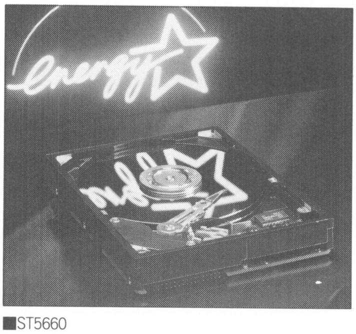
富士通㈱（03-5473-1261/1262）は，同社が開発したMRヘッドを搭載した2.5インチ/3.5インチハードディスクドライブ3シリーズ6機種を発売する．同シリーズは，2.5インチサイズで最大528MBを実現した「M270Xシリーズ」，3.5インチサイズ，1インチ厚で1GBの「M292Xシリーズ」，3.5インチサイズで最大3GBの「M29XXシリーズ」．全機種SCSI-2対応，M270XシリーズのみIDEもサポートする．

MicroDevice

Seagete，EnergyStar準拠の3.5インチハードディスクなど

日本シーゲイト㈱（03-5476-5821）は，厚さ0.75インチで545MBの容量の3.5インチハードディスク「ST5660A（IDE）/ST5660N（SCSI-2）」と，ポータブルコンピュータ向けの2.5インチハードディスク「ST9550AG（455MB）/ST9385AG（341MB）」を発表した．

「ST5660」ファミリは，米環境保護局（EPA）のEnergy Star Computer Program準拠の省エネ設計を採用．今後，国際的な広がりが予想されるグリーンPCへの採用が規定される．OEM用サンプル価格はST5660Aが495ドル，ST5660Nが545ドル，ST9550AGが595ドル，ST9385AGが495ドル．



■ST5660

Hardware News

Hardware News

GATEWAY2000，PCI採用 Pentiumプロセッサ搭載機

日商岩井㈱(03-3588-2419)は，Pentium/60MHzを採用し，PCIスロットを3基搭載したGATEWAY2000社の「GATEWAY2000 P5-60」を販売する．同機種はメインメモリ16MB，HDD 528MB，3.5インチFDDを内蔵．ビデオカードにATI ULTRA XLR(2MB)を採用し，最大1,280×1,024ドット/256色表示が可能．価格は59万8千円．

Hardware News

富士通，TFT カラー液晶採用 i486SX搭載ノートパソコン

富士通㈱(03-3646-0816)は，9.4インチTFTカラー液晶を採用し，CPUにi486SX/25MHzを搭載したノート型パソコン「FMR-50NL/T」を発売する．同製品は，JEIDA/PCMCIA仕様準拠のPCカードをサポート，TypeⅡを2枚内蔵可能となっている．画面表示は640×480ドット表示に対応している．また，120MBハードディスク内蔵タイプには，Windows 3.1プリインストールモデルを用意．メモリは標準で2.75MB，Windowsインストールモデルでは6.75MB．価格はFDDモデルが44万8千円，80MB HDモデルが54万8千円，120MB HDモデルが57万8千円，120MB Windows 3.1プリインストールモデルが64万8千円となっている．

Hardware News

エプソン，新エスパーレーザープリンタ

セイコーエプソン㈱(03-3377-3500)は，エスパー・レーザープリンタの新機種として，A3対応プリンタ「LP-8000S」とA4対応プリンタ「LP-1500S」を発売する．両プリンタともCPUの高速化をはかり，出力時間の短縮化を実現している．また，解像度強化機能RITにより600dpi相当の印字を実現している．明朝体，ゴシック体のアウトラインフォントを標準装備．価格はLP-8000Sが29万8千円，LP-1500Sが14万8千円．

Hardware News

COMPAQ，Pentiumプロセッサ搭載デスクトップなど

コンパック㈱(03-5210-3891)は，Pentiumプロセッサ搭載のデスクトップPC「DESKPRO XE 560」，Energy Star Computer Programに準拠した「DESKPRO XE 466」，ミニタワー型PC「ProLinea MT 4/66」を発売する．

「DESKPRO XE 560」は，CPUにPentium/60MHzを搭載し，64ビットTriFlex PCアーキテクチャを採用，「DESKPRO XE 466」は，CPUにi486DX2/66MHzを搭載，Pentiumオーバドライブプロセッサ(P24T)へのアップグレード可能．両機種ともメモリ8MB，Qvisionグラフィックスコントローラをローカルバスに直結している．「ProLinea MT 4/66」はコンパクトなミニタワーデザインを採用，ISAスロット×5とドライブベイ×5を装備，CPUにi486DX2/66MHz，メモリ標準4MB，HDD 340MBを搭載している．



■LP-1500S

また，同社は，ネットワーク対応でB4印字が可能なインパクトプリンタ「エスパー・インパクト VP-6000」を発売する．同プリンタは通常印字で150文字(漢字)/秒の速度を実現している．またオプションにより，最大2種類のインターフェースの自動切替などが可能．価格は64万8千円．

Hardware News

YHP DeskWriter550C値下げ

横河・ヒューレット・パッカード㈱(03-3335-8333)は，Macintosh向けカラーインクジェットプリンタ「HP DeskWriter550C」の価格を14万8千円から10万8千円と，約27%の値下げを行った．

また，P24TによりPentiumプロセッサへのアップグレードが可能なZIFソケットを標準装備している．価格は「DESKPRO XE 560」の「モデル270w」(ハードディスク270MB)が48万8千円，「モデル525w」(ハードディスク525MB)が54万8千円，「DESKPRO XE 466」の「モデル270w」が35万8千円，「モデル525w」が41万8千円，「ProLinea MT 4/66」が33万8千円．



■DESKPRO XE 560

Hardware News

キヤノン，世界最小，最軽量カラーフラットベットスキャナ

キヤノン㈱(03-3455-9540)は，A4サイズ/フルカラー対応で，フラットベットタイプとしては世界最少・最軽量のイメージスキャナ「IX-4015」を発売する．

同製品は，24ビットフルカラースキャナでありながら，286(W)×405(D)×79(H)mm，約6kgの軽さを実現した．さらに，A4フルカラーで約20秒，A4モノクロで約10とスキャニングスピードの高速化もはかられている．解像度はカラー:400dpi×800lpi，モノクロ:400dpi×1200lpi．本体価格は13万8千円，Macintosh用I/Fキットは4万円．

Hardware News

SONY，98対応マルチスキャンディスプレイなど

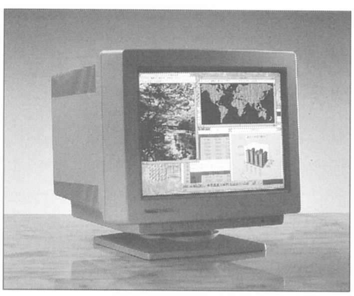
ソニー㈱(03-5448-3311)は，17インチディスプレイ2機種と14インチディィスプレイ1機種を発売する．「GDM-17SE1」は，水平周波数31.5～82kHzに対応し，色調の微妙な調整も前面のボタンで行うことができ，高い精度が要

求されるCAD業務分野向けとなっている．また，米環境保護局（EPA）のEnergy Star Computer Program準拠のパワーセービング機能を装備．「CPD-1738」は水平周波数24～58kHzに対応し，最大1,024×768ドットまでの表示が可能．両製品ともAGピッチ0.25mmのファインピッチトリニトロン管を搭載．また，「CPD-1438」は水平周波数24～58kHz，AGピッチ0.25mmの14インチファインピッチトリニトロン管を採用．3製品ともマイクロプロセッサーによるデジタルコントロールシステムを搭載．VCCI 2種にも適合している．価格は「GDM-17SE1」が34万8千円，「CPD-1738」が18万8千円，「CPD-1438」が9万8千円．

Hardware News

ナナオ，ErgoCoat採用の廉価版T560i・J

㈱ナナオ（0762-74-2434）は，17インチマルチスキャンディスプレイ「FlexScan T567」を発売する．同製品は一般のトリニトロンCRTよりも高品位なフラットタイプトリニトロンCRTを採用．走査周波数は，水平30～82kHz，垂直55～90Hz，電力消費を押さえるパワーマネージャ機能を搭載するなど中身はT560i・Jと同じだが，F557などに採用された，外光反射を押さえるErgoCoatを装備することにより，高品位な画質を保ちながら31万円という価格を実現している．

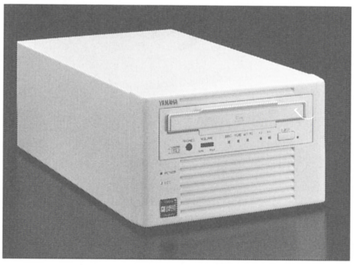


また，同社は1月末日まで「FlexScan 98キャンペーン」として，98ユーザに対し，「FlexScan T660i・J/T560i・J/F347」の各ディスプレイに，98ノーマルモードでも表示が可能になる「ErgoVerter 98」を付属させる．

Hardware News

YAMAHA，4倍速のCDレコーダ

ヤマハ㈱（0539-62-3125）は，4倍速でデータの書き込みができる追記型CD書き込み装置「CDE100」およびハードディスク内蔵モデル「CDE100H10」を'94年2月から発売する．同製品は，CD-R（Recordable）ディスクを使用し，CD-ROM，音楽用CD（CD-DA）の各フォーマットを4倍速で読み書き可能．CD-ROM1枚の記憶容量約600MBのデータを約16分で記録できる．また，制御用のライティングソフトウェアも同時発売される．価格はCDレコーダー「CDE100」が60万円，「CDE100H10」が90万円，Macintosh用ライティングソフトウェアは20万円，PC/AT用ならびに98用は25万円．



Hardware News

オリンパス，1.3GB容量の5.25インチMOドライブ

オリンパス光学工業㈱（03-3340-2020）は，容量1.3GBの5.25インチ光磁気ディスクドライブ「MOS520S」を発売する．回転速度3,000rpm，転送速度2MB/sec，シークタイム37msを実現．インターフェースはSCSI-1/2に対応．価格は43万円．

Hardware News

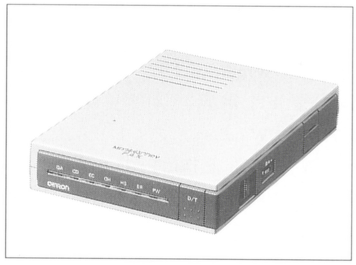
東芝，1インチハイトの内蔵型CD-ROMドライブ

㈱東芝（03-5484-5713）8は，内蔵型CD-ROMドライブで初めて高さを1インチとしたOEM専用ドライブ「XM-4101B」を発表した．同ドライブは転送速度300KB/sの倍速ドライブで，CD-ROM XAやPhotoCDのマルチセッションなどに対応している．サンプル価格は7万円．

Hardware News

オムロン，Windows対応FAX/DATAモデム

オムロン㈱（03-5488-3221）は，FAX/DATAモデム「MD96XT10V-W」，「MS144XT10V-W」を発売する．同製品は，それぞれ9600bps，14400bpsのFAX/DATA通信が可能で，FAX通信用ソフト「MDFAX」とデータ通信用「MDATA」で構成されるWindows対応通信ソフト「MDFAX for Windows」が標準添付される．動作環境はPC-98シリーズおよびその互換機など．CPUは80386 16MHz以上，OSはMS-Windows 3.1，メインメモリ4MB以上（推奨8MB），ハードディスクの空き容量1.5MB以上となっている．価格は「MD96XT10V-W」が4万6,800円，「MD144XT10V-W」が5万2,800円．



Hardware News

U.S.Robotic，RockwellのV.fast対応製品を採用

米U.S.Robotics社は，Rockwell社の「V.Fast Class」技術を同社の製品に採用することについて合意したと発表した．これにより同社ではITU-T（旧CCITT）のV.34決定までの間V.fast対応のモデムを出荷することになる．

Hardware News

Hayes，PCMCIA対応のFAXモデム

米Hayes社は，PCMCIA Type II 対応のFAXモデム「OPTIMA 144+FAX144 for PCMCIA」を発表した．同モデムは

V.32bis/V.42bisなどに対応している．価格は399ドル．

Hardware News

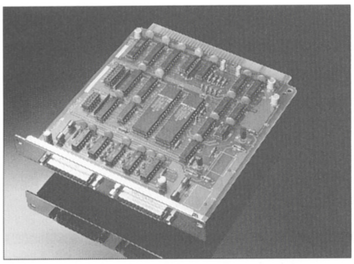
ATT Paradyne, マルチメディアモデム

米ATT Paradyne社は，通話中にもデータの送受信が可能な「ATT DataPort 2001 Multimedia Communicator」を発表した．同モデムは通話中に最大4,800bpsでのデータ転送が可能なほか，通常のFAXモデムとして最大14,400bpsの通信も可能．価格は600ドル以下の予定．

Hardware News

メディアインテリジェント, 98用高速R-232C拡張ボード

メディアインテリジェント㈱(0424-99-1800)は，EPSON PCシリーズおよびPC-9801シリーズ対応のRS-232C拡張ボード「RSB-384」を発売する．同機種は16BのFIFOバッファを持つRS-232Cポートを二つ持ち，最大1.152Mbpsの転送速度で使用可能．またWindows用ドライバソフトが付属する．価格は2万9,800円．



Hardware News

アイテック, プリンタポート接続用HDD

アイテック㈱(0720-68-5331)は，プリンタポートに接続する，携帯性に優れ，衝撃にも強いハードディスク「PHd」シリーズを発売する．同製品は，シンガポールのK.T.TECHNOLOGY社の製品で，PC-AT互換機のプリンタポートに接続して使用する．データ転送速度は400KB/sec．動作時10G，非動作時150Gの衝撃に耐える．価格は以下の通り．

PHd080	80MB	7万7千円
PHd120	120MB	9万円
PHd205	205MB	11万5千円

Hardware News

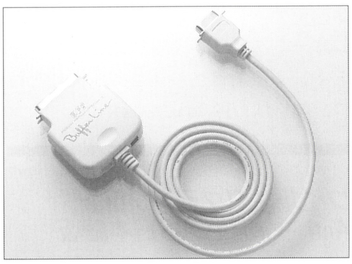
緑電子, 98NOTE用SCSIアダプタなど

緑電子㈱(044-989-7633)は，NEC 98NOTEシリーズ用のSCSIアダプタ「MDC-554NA」を発売する．価格は2万4,800円．また，EPSON NOTE用内蔵ハードディスクパック「びるとE」シリーズも発売する．価格は7万7千円(120MB)/9万7千円(200MB)/13万7千円(340MB)．

Hardware News

八戸ファームウェアシステム, 新しい形のプリンタバッファ

八戸ファームウェアシステム㈱(011-702-6681)は，従来の大きくて高価なプリンタバッファをプリンタケーブルのコネクタ部分に収めた新しいコンセプトのプリンタバッファ「Buffer Line」を発売する．同製品は，低価格ながらバッファ容量1MBの大容量をもち，プリントアウトの際のアプリケーション開放時間が大幅に短縮される．PC-9800/EPSON PCシリーズデスクトップ用/ノート用があり，価格は1万9,800円．



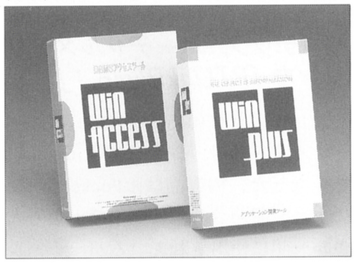
Software News

Software News

日商エレクトロニクス, WinPlus, WinACCESS

日商エレクトロニクス㈱(03-3544-8274)は，SpinnakerSoftware社のWindowsアプリケーション開発ツール「WinPLUS」とDBMSアクセスツール「WinACCESS」の日本語対応版を発売する．

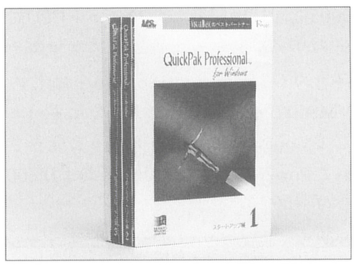
WinPLUSは，MacintoshのHyperCardで使用されているHyperTalkと同様のスクリプト言語(PPL:Plus Program Language)を持ち，HyperTalkと互換性がある．データベースフィールドやボタンでは，スクリプトの自動生成も可能．また，WinACCESSはWinPLUSにDBMSアクセス拡張機能を追加したもので，販売はWinPLUSと一体で行われる．WinACCESSはDBMSへのアクセスを行うためのフロントエンドツールとして，現在市販されているほとんどのDBMSへのアクセスをサポートしている．価格はWinPLUSが9万9,800円，WinACCESSが14万8千円(WinPLUSを含む)．



Software News

文化オリエント, QuickPak Professional

文化オリエント㈱(022-378-7106)は，米Crescent Software社のVisual BASIC用多目的ツールライブラリの日本語版「QuickPak Professional for Windows」を発売する．価格は5万3千円だが，発売記念キャンペーンとして12月31日まで4万3千円で販売する．



Software News

ビービーエス, 制御系ソフトのCコードジェネレータ

㈱ビービーエス(03-3496-1544)は，パソコンで動作する制御系ソフトのCコードジェネレータ「VisSim/C-Code」を発売．同製品は米Visual Solutions社

開発の非線形ダイナミックシステムのシミュレーション用ソフト「VisSim」で構築したブロック線図に対応するCのソースコードを自動生成することができる．動作環境はMS-Windows 3.0以上．価格は15万7千円．

Software News

ソーテック，バーチャルメモリリンカ「RTLinkPlus6.1」

㈱ソーテック(045-661-7360)は，バーチャルメモリリンカ「RTLinkPlus」を発売した．同製品は米PocketSoft社によって開発され，世界中で十数万人のユーザに使用されているバーチャルメモリ&オーバレイリンカ．同製品を使用することにより，ソースコードの変更をせずに大規模なプログラムの実行が可能になる．

Software News

マイクロウェア，PowerPC用OS-9000

マイクロウェアシステムズ㈱(03-3257-9003)は，米Microware Systems社がIBM最新RISCプロセッサPower PCに組み込まれたOS-9000を発表した．OS-9000は同社が開発した32ビットCPU対応のリアルタイムマルチタスクマルチユーザOSであり，386/486をはじめVM8600，各社RISCチップをサポートしている．今回サポートすると発表されたPowerPCは400シリーズおよび600シリーズ．

Software News

コムユース，Windows用FAXソフト

コムユース㈱(0425-26-1550)は．EIA-578 Class1採用のFAXモデムに対応した「High-FAX」Windows対応版を発売する．同製品はFAX送受信やその周辺業務をパソコンで処理することができる．また，電話帳機能を持っているためデータベースソフトがない場合でも顧客管理が行える．価格は2万8千円(予価)だが，新発売キャンペーンとして発売日より2カ月間，定価の半額セールが実施される予定．

Software News

ブラザー，Windows用FAXソフト

ブラザー販売㈱(052-263-5882)は．Windows上でFAX送受信が可能なFAX通信ソフト「CB-FAX Pro」を発売する．同ソフトはClass 1/2のFAXコマンドに対応．また，バックグラウンドでFAXの自動受信が可能．価格は1万4,800円．

Software News

クレッセンド・カンパニー，Windows対応住所録データベース

クレッセンド カンパニー(043-222-9801)は，Windows用多機能住所録データベース「MAGIC ADDRESS(マジックアドレス)」を発売する．同製品は会社，商店での顧客管理ソフトとしても使用でき，条件検索，抽出，データ記録が可能．各ファイルにはメモの入力が可能．1ファイルデータにつき6種類までの電話番号を登録することができ，モデムをつなぐと指定の電話番号にオートダイアルする機能や毛筆フォントによる葉書への宛名印刷機能を持つ．価格は1万9千円．

Software News

ウイルスワクチンソフトAnti -Virus PLUS日本語版

サイトー・オフィス・クリエイティブ㈱(03-3578-0491)は，IRIS-SOFTWARE AND COMPUTERS社の開発した「Anti-Virus PLUS」を日本語に対応させ，「ワクチンコンバット」の製品名で発売する．同製品は同一の実行ファイルで日本語・英語の両方に対応しており，Windows版(英語メニュー)も含まれている．また，コンピュータウイルスの検出，感染ファイルの修復，再構築，ウイルスの監視，システムファイルなどが前回起動時と変更がないかどうかの整合性のチェックなどの機能を持つ．対応機種はIBM PC/AT 互換機のみ．価格は9,800円．

Software News

リンク，法人向けウイルス対策サイトライセンスパック

㈱リンク(03-3493-5850)は，ウイルス対策の法人契約をパッケージ化して提供する．「サイトライセンス契約」は，自社のパソコンに対してウイルス検査したい場合の契約で，期間は2年間．価格は20万円(25台契約の場合)．「サービスインダストリーライセンス契約」は，他社のウイルス検査を代行する場合や，自社製品のメンテナンス業務などの場合の契約で，期間は2年間．価格は25万円(1～5人のメンテナンス要員の場合)．

Software News

アドビシステムズ，日本語Windows版Photoshop 2.5

アドビ システムズ ジャパン(03-3437-8950)はWindows版「Adobe Photoshop 2.5.1J」を発売する．同製品はグラフィックイメージ処理ソフトウェアで，Macintosh版とファイルフォーマットのバイナリ互換性もある．また，TrueTypeおよびATM Type1フォントを完全にサポート．価格は15万5千円．なお，英語バージョンのWindows版「Adobe PhotoShop 2.5」の正規登録ユーザは4万円で日本語版にアップグレード可能．

Topics

Topics

Intel，インテルベリフィケーションプログラム実施

インテルジャパン㈱(0298-47-8511)は，i486CPU搭載パソコンを対象にアップグレード設計の検証および認定を行う「インテルベリフィケーションプログラム(Intel Verification Program)」を開始した．

同プログラムは，将来発売されるPentiumプロセッサベースのオーバードライブプロセッサを利用したアップグレードに対応しているとされるi486CPU搭載パソコンが，それに必要な最低限の設計基準を実際に満たしているかを検査し，すべての基準を満たした機種を認定するもの．

INFORMATION
BUG & Aftercare
Events Presents etc…

Bug & Aftercare

■11月号特集1「FAXモデムの内部構造」P.38左段17行目の「互いに直交する(180度位相差がある)キャリア…」は,「互いに直交する(90度位相差がある)キャリア…」の誤りです.お詫びして訂正いたします.

■12月号「短期集中ロードテスト ZEOS Contenda」P.32,のContenda輸入取り扱いショップ「くまくま」の住所に誤りがありました・正しくは,

〒870　大分市高崎4-15-6

です.お詫びして,訂正いたします.

■12月号P.124～140「PC-9801RA2のバンクメモリをWindows3.1のDOS互換ボックス使用する実験レポート」で紹介したWindows3.1でI・Oバンク方式のメモリをアクセスできるRAMディスクドライバに関して,㈱アイ・オー・データ機器より「同様のドライバは,現在開発中であり,完成次第,MEMORY SERVERに添付する予定」との回答をいただきました.

Events

■グループウエア・ソリューション '94

グループウエア・ソリューション実行委員会および日本経済新聞社は,グループウェアを日本で普及・啓蒙し,日本の産業界の活性化と情報通信業界に新風をもたらすことを目的として,カンファレンス,ショーケースおよびワークショップから構成される国際会議「グルーウエア・ソリューション'94」を'94年2月3日㈭～4日㈮の2日間,日本コンベンションセンター(幕張メッセ)および幕張プリンスホテルで開催する.

会期:'94年2月3日㈭～4日㈮
会場:日本コンベンションセンター(幕張メッセ),幕張プリンスホテル
料金:カンファレンス(2日間)　8万円
ショーケース,ワークショップ　　無料
問い合わせ先:グループウエア・ソリューション'94事務局
〒104 中央区築地2-11-3ヒロシゲビル
☎03-5565-4906 / FAX 03-5565-4908

■親子を対象としたパソコンセミナー

NECは,'94年3月まで,全国で合計1万組以上の親子を対象に,パソコンを使用した家庭での学習方法や家族全員で楽しめる活用法などを体験できる大規模なパソコン活用セミナーを企画.協賛販売店とともに「楽しく勉強,まじめに遊ぶ　親子で楽しくパソコン活用セミナー」の名称で開催する.同セミナーは,販売店のセミナールームなどを会場とし,全国約200カ所で1,500回,土,日曜日を中心に開催される.受講対象者は,小学4～6年生を中心とした親子ペア.11月,12月のセミナー内容は,①学年別問題を選択,クイズ形式での学力試し.②クリスマスカード,年賀状などに使用する図案の選択とぬり絵.

なお,応募方法やお問い合わせは,最寄りのセミナー実施販売店へ.

Presents

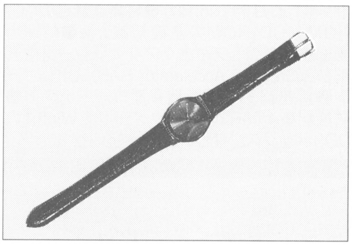
1993年11月号プレゼント当選者

❶LANGUAGE/OS
横浜市　野々村一志,西宮市　奥村　勇
❷MEGA MEDIA 1
綾瀬市　松本好章,小松市　白栄隆司
❸マクロキー入力コンバータ「マッキー」
山梨県　西川雄一,橿原市　澤田弘樹
東大和市 風間利則
❹UnixWareキャップ
松戸市　川口 誠
❺CREATIVE TECHNOLOGYテレカ
札幌市 小笠原和宏,淀川区 内海敏彦
❻FOOTWORK RACING TEAMテレカ
墨田区　松本圭,西条市　河端和行
兵庫県　花澤豊丸
❼UG DESKTOPテレカ
尼崎市　明石久夫,埼玉県　上 貴幸
大分市　足立 敦
❽女プログラマーは朝臥
横浜市　万野寿和,奈良県　和田康男
熊本市　池田政信
❾コモエスタアミーガ!
調布市　倉山 潔,長崎市　黒崎博之
八王子市 芦田秀規　(以上　敬称略)

今月のPresents

❶Cyrix腕時計

サイリックス㈱より,ロゴ入り腕時計を1名の方に.



❷QuickPak Professional forWindows

文化オリエイト㈱より,Windows版Visual BASIC用アドインソフト「QuickPak Professional for Windows」(3.5インチ1.2MB)を2名の方に.

❸CROSSボールペン

㈱ジャストシステムより,CROSSのロゴ入りボールペンを1名の方に.

❹3Com名刺入れ

3Com社の日本法人スリーコム・ジャパン販売㈱の設立を記念して,3Comのロゴ入り名刺入れを1名の方に.

❺IBM ThinkPad 220徹底活用ブック

㈱ビー・エヌ・エヌより,「IBM ThinkPad 220徹底活用ブック」(3.5インチディスク入り)を2名の方に.

今月のプレゼントをご希望の方は,本誌とじ込みの愛読者カードでお申し込みください.また,本誌記事に関するアンケートには,必ずお答えください.申し込み締切は,1月18日㈫(必着)です.

BACK NUMBER

バックナンバー案内
地域別バックナンバー常備店

■直接，書店にお申し込みになるか，弊社営業部バックナンバー係まで，代金分の「切手」を添えてお申し込みください（定価は消費税込み．送料弊社負担）．お**申し込みの際は，必ず送付先と希望月号を明記**した紙を同封してください．お申し込み後，2週間経っても届かない場合は，営業部までお問い合わせください．

〈申し込み先〉
〒160 東京都新宿区愛住町8番地8
㈱技術評論社　営業部
バックナンバー係
☎03-3225-2300



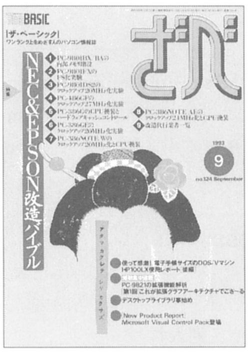
'93年**12**月号(No.127)●特集「魅惑のCD-ROM」CD-ROMの基礎知識，他●特別企画：続々・グラフィックアクセラレータ徹底解剖・Graphics Accelerator HA50試用レポート●新連載：AT互換機長期ロードテスト●一般記事：98R2のバンクメモリをWin3.1のDOSボックスで使う，他(定価：600円)



'93年**11**月号(No.126)●特集「闘うFAXモデム」FAXモデムとFAXソフトの最新動向，他●特別企画：続・グラフィックアクセラレータ徹底解剖～PowerWindow928試用レポート他●一般記事：新98E解体新書／98NS/E，NC，NS/TのレジュームBIOSバグ対処法／VB3.0 Review他(定価：600円)



'93年**10**月号(No.125)●特集「続・3.5インチMO研究序説」98/Mac/ATで使う3.5インチMO，他●一般記事：ThinkPad220の期待と誤算～電池はそんなに長持ちしない／Rewriting Win3.1 Device Drivers with TP for Windows／Cx486DRx^{2}試用レポート／J-Scan+PC-Transer/je他(定価：600円)



'93年**9**月号(No.124)●特集「NEC & EPSON改造バイブル」PC-9801BX/BAの内部メモリ増設，他●重点特別企画：Cyrix突撃インタビュー●一般記事：デスクトップライブラリ事始め／IBM DOS J5.02/V vs MS-DOS 5.0/V●連載：(短期集中連載)PC-9821の拡張機能解析(1)，他(定価：600円)



'93年**8**月号(No.123)●特集「一太郎Ver.5を斬る!!」ATOK8を徹底解剖する，他●特別企画：PC-9801改造バイブル●一般記事：HP100LX使用レポート(前編)／30行計画でATOK6/7/8を使おう!!／新・98MULTi解体新書／文字ビジョン2使用レポート●連載：Advanced Assembler386(4)，他(定価：600円)



'93年**7**月号(No.122)●特集「グラフィックアクセラレータ徹底解剖」グラフィックアクセラレータボードの現状と内部構造，他●一般記事：FiLo試用レポート／30行計画を統合しよう！／その後のロシア電脳事情(前編)●新連載：傍晩好像聴到香港的鍵盤聲，他(定価：600円)

●その他のバックナンバー在庫(5月号=特別定価690円を除き，すべて定価600円)
・'93年6月号(No.121)／特集「Windows3.1を使いこなせ」／特別企画：またCx486SLC/DLC／連載：電脳辞書の国語学(最終回)，他
・'93年5月号(No.120)／特集「パソコンパワーアップ大作戦」／特別企画：キーマンが語る／一般記事：98MATE解体新書，他
・'93年3月号(No.118)／特集「プログラミング革命！　Visual BASIC」／一般記事：286用Cx486SLCボード徹底比較，他
・'93年1月号(No.116)／特集「DOS/Vの拡張とHiText」／新連載：香港電脳界探訪指南／ATOK6/7を30行計画に対応する，他

●バックナンバー品切れ情報
創刊号～'92年12月号，'93年2月号，'93年4月号は，在庫がございません

PROGRAM SERVICE 掲載プログラムサービス

'94年1月号ディスクサービス申込用紙

郵便番号　　　　メディア

住所（アパート，マンションなどの場合は部屋番号までご記入ください）

お名前

様

■本サービスは，入力の手間を省くため，またCやアセンブラなどの言語で書かれたプログラムを，それらの言語処理系を持っていない方でも使えるようにという主旨で，ソースファイルおよび実行ファイルを提供するものです．そのため，基本的にはマニュアルなどは添付しておりません．あらかじめご了承ください．

掲載時のプログラムをそのまま収録してありますので，掲載後に発見されたバグなどにつきましては，随時，INFORMATIONのページの「BUG & AFTERCARE」欄に掲載致しますので，ご覧ください．

なお，収録プログラムは，弊社主催のBBS，Jupiter(Ganymede)-Net(会員制・有料)で入手することも可能です．

〈申し込み方法〉

●価格

1,500円(必ず切手で)

●供給メディア

5"2HD or 3.5"2HD

●サービス期間

'94年2月末日まで

●申し込み方法

必要事項を記入した申込用紙・切手1,500円分を同封して，弊社「管理部ざべディスクサービス係」宛にお送りください．受け付け後，1〜3週間程度でディスクを発送致します(**申込用紙は右上のものを切り取るか，コピーしてご使用ください．そのまま宛名として使いますので，楷書ではっきりとご記入ください**)．

〈ご注意〉

収録プログラムの内容などに関するご質問は，本誌編集部(**☎03-3225-3295/月曜日〜金曜日/14:00〜17:00**)までお問い合わせください．

掲載プログラムサービスのバックナンバーは，サービス期間を過ぎたものでも在庫がある場合に限り，受け付けておりますが，確実に手に入れるためには，サービス期間中にお申し込みくださるようお願い致します．

■今月の収録プログラム

●創刊号から128号(本号)までの総目次

バックナンバーは全国の書店でお求めになれます

〈北海道地区〉
★旭屋書店　札幌店
〒060　札幌市中央区南三条西4丁目　アルシュB2　☎011-241-3007
★YES　電脳館
〒060　札幌市中央区南二条西3丁目　YES 5F　☎011-214-2850
★室蘭工業大学生協
〒050　室蘭市水元町36-8　☎0143-44-6060
〈東北地区〉
★金港堂ブックセンター
〒980　仙台市一番町3-11-15 フォーラス6F　☎022-223-0979
〈信越地区〉
★紀伊国屋書店　新潟店
〒950　新潟市万代1-3-30 万代シルバーホテル2F　☎025-241-5281
★うつのみや書店　片町店
〒920　金沢市片町2-1-7　☎0762-21-6136
〈東京地区〉
★三省堂書店　神田本店　5F
〒101　千代田区神田神保町1-1　☎03-3233-3314
★書泉グランデ　5F
〒101　千代田区神田神保町1-3　☎03-3295-0011
★書泉ブックマート　B1
〒101　千代田区神田神保町1-21-6　☎03-3294-0011
★大盛堂書店　5F
〒150　渋谷区神南1-22-4　☎03-3463-0511
★芳林堂書店　池袋本店　5F
〒171　豊島区西池袋1-11-7　☎03-3984-1101
★T·ZONE　7F　ブックゾーン
〒101　千代田区外神田4-4-1　☎03-3257-2660
★紀伊国屋書店　新宿本店　4F
〒160　新宿区新宿3-17-7　☎03-3354-0131
★くまざわ本店　4F
〒192　八王子市旭町2-13　☎0426-25-1201
〈京浜地区〉
★有隣堂　横浜西口店
〒200　横浜市西区北幸1-6-B1　トーヨー地下街　☎045-311-6265
★文学堂　2F
〒210　川崎市川崎区砂子1-3　ニューハヤトビル　☎044-244-1251
★藤沢リブロ
〒251　藤沢市南藤沢3-1　西武百貨店6F　☎0466-27-0111
〈関東地区〉
★船橋リブロ
〒273　船橋市本町1-2-1　西武百貨店10F　☎0474-25-0111
★新星堂
〒277　柏市柏1-2-31　新星堂カルチェ5　3F　☎0471-64-8551
★川又書店　駅前店
〒310　水戸市宮町2-2-31　☎0292-31-0102
★多田屋　千葉セントラルプラザ店
〒280　千葉市中央3-17-1　セントラルプラザ5F　☎0472-24-1333
〈東海地区〉
★三省堂書店　名古屋店
〒450　名古屋市中村区名駅1-1-2　名古屋ターミナルB2　☎052-562-0077
★パソコンショップΣ
〒460　名古屋市中区大須3-30-37　☎052-251-8334
★ブックセンター名豊
〒448　刈谷市新富町3-43-1
〈関西地区〉
★ジュンク堂書店
〒650　神戸市中央区三宮町1-6-18　☎078-392-1001
★アバンティ　イズミヤブックセンター
〒601　京都市南区東九条西山王町31　アバンティ6F　☎075-682-5031
★駸々堂バージョン99
〒534　大阪市都島区東野田町2-1-38　京阪モール北館4F　☎06-353-4011
★ニノックス・コア　日本橋店1F
〒556　大阪市浪速区日本橋西1-1-3　☎06-647-2038
〈中国・四国地区〉
★紀伊国屋書店　岡山店
〒700　岡山市中山下2-2-1　☎0862-32-3411
★紀伊国屋書店　広島店
〒730　広島市中区基町6-27　広島センタービル6F　☎0822-25-3232
★宮脇書店
〒760　高松市丸亀町4-8　☎0878-51-3733
〈九州地区〉
★紀伊国屋書店　福岡店
〒810　福岡市中央区天神1-11-1　天神コアビル6F　☎092-721-7755
★紀伊国屋書店　熊本店
〒860　熊本市下通り1-7-18　☎0963-22-5531

●上記書店では，当社パソコン関連図書も取り揃えています

言いたい放題♪

♠'93年11月号の愛読者カードから，ご意見・ご感想などを紹介します．



FAXモデム

モデムが安くなったよー．それも14,400bps．この間9,600bpsのFAXモデムを買ったばかりなのに．これから買うときには，先をよ～く見通した上で買いましょう．しかし，最近ハードウェアはこんなのばかりですね．よいのか，悪いのか．トホホ….

【東京都　ともとも】

TB 海の向こうでは，すでにV.fast(28,800bps)のモデムが500ドル前後で買えます．日本のメーカーにも頑張ってもらいたいものです．

FAXモデム特集に松下のTO-703Bが大きく扱われていますが，あのようなマイナーな機種に肩入れするのはどうかと思う．

【塩尻市　mike】

TB 松下のTO-703Bですが，とりたててマイナーということはないと思いますけど…….

今月の特集で感じたこと，というよりも最近のモデム関連の記事(他誌も含めて)で感じたことは，必ずといっていいほどロックウェル・インターナショナルの社名が出てくることである．ロックウェルといえば航空ファンおよび軍事オタクにとっては有名な企業である．わりと最近(そうでもないか)では，ステルス爆撃機のはしりといわれた(あくまでもはしりであって本格的なものではない)B-1“ランサー”が有名である．たしかスペースシャトル1号機のエンタープライズもロックウェルじゃなかったかな．他のスペースシャトルもそうだったような気がするが，定かではない．もっと昔には，レシプロ戦闘機の最高傑作といわれたP-51“ムスタング”なんてのも作っておりました(当時の社名はノースアメリカン)．それが今ではモデム用チップを作っているとは．これも昨今の軍縮のせいなんでしょうか．前から半導体部門があることは知っていましたが．これからはV.fastのことをムスタングと呼びましょうか．

【岩見沢市　佐藤C作】

CD-ROM

CD-ROMのプレゼントやソフトの広告をよく見かけるようになりました．先日，会社の人と話していたら，2人ともCD-ROMを持っていると聞いて驚きました．我家はパソコン用はおろか，音楽用のCD-ROMもまだありません．あの，虹色の輝きにあこがれています．ところで，パソコンソフト用のCD-ROMを音楽用のプレーヤーにつなぐとどうなるのでしょうか．ピーとかブーブーとか警告音か音声で何かメッセージ出してくれないのですか?(逆はソフトにより，音楽の聴けるCD-ROMドライブとか何かあったような気がしますが……)

【新居浜市　光る虹色】

池袋のサンシャイン(?)であったCD-ROM展示即売会に友人を派遣したところ，10時の開場前から長蛇の列で，先着500名へのプレゼントCDはもらえなかったとのこと．Nautilusのバックナンバーなどを購入してきてもらったが，比較的最近のバックナンバーはすでに売り切れだったとのこと．すごい勢いで普及していることがひしひしと伝わってきた．

【福岡市　加藤勝憲】

TB 12月号のNautilusCDサンプル版プレゼントにも多数のハガキが寄せられています．本当にみなさんCD-ROMドライブを持っているんでしょうか？それとも単にタダでもらえるものはもらおうということなのでしょうか？

ここだけの話

ここだけの話しで，あまり詳しく書けませんが，コンピュータの話で，笑うに笑えない話があります．つい最近まで2年がかりでやっと本番に入ったコンピュータシステムの仕事をやっていました．全部で約2億円のシステムで，千葉の某ホテルのものです．営業が始まって，人の入りはまあまあだと思っていましたが，1週間が経ったころにこんな事がありました．ホテルの中のお店の売上がホストコンピュータに送られてこなかったのです．お店が営業しているのに送ってこないとは，プログラムのバグではないかと考え，一生懸命調べました．いくら見ても分からないので，詳しく話を聞いてみると，お客さんが一人もこなかったとの事！　これにはおどろいてしまったのでした．泊まったお客さんが1日で6人とかで，コンピュータなんか必要のないくらいです．2，30万円くらいのパソコンを2，3台置けば，十分に用が足りてしまうではないか！　2年頑張ったのに，すごくショックです！　信じられません．

【横浜市　NSR250R+横浜ナンバー】

拝啓メーカー様

私の勤務する会社では，ページプリンタとしてメモレックステレックス社の218Xシリーズを使っています．このプリンタはカシオのOEMです．最近ちょっとした問題がありました．Windowsのバージョンは3.1でTrueTypeフォントを使おうとすると，フォントリストに出てこない．調べてみると，プリンタドライバが3.0用のものであることが原因だった．さっそくメモレッ

クステレックスに電話で問合わせをすると，「3.1用ドライバは作らない．3.1に対応したプリンタを作ったからそちらを買って欲しい」とびっくりするような返事．で，プリンタとして新しく買ったのはエプソンのLP-8000になった．

【海老名市　平始旬】

PC-98RL+PC-9801-86またはPC-9821CeでMS-Golfをプレーすると，チップ音やパター音が正しく再生できない．0.02秒～0.03秒の音声(WAV)だとだめで，頭に0.02秒のブランクを付加すると聞こえるようになるのだが，NECは例によって認めようとしないし….Wired for Soundのメーキングマインスイーパーの時計の音も小さすぎて聞こえないし．誰かドライバを修正してもらえませんか．

【仙台市　久松与悦】

ソフト会社のサポートは冷たいと思っていた．とあるソフトが手持ちのプリンタに対応していないので，モンクを言ったら，「そのプリンタには対応できない．次期バージョンアップの際考慮する」と言われた．そのプリンタはN○Cの最新レーザプリンタだった．それから，8カ月すぎてからTelがあった．「以前のサポートの回答に誤りがあった．そのプリンタには対応している．最新版に書き換えてやるから，マスターディスクを返送しろ」とのことだった．最新版に書き換えてもらい，快適に使用している．さすが，大会社ア○キーのサポート係！でも，8カ月はちょっと遅すぎませんか?

【宇佐市　T.C】

拝啓ユーザ様

NM Windowsドライバに対する小澤重人様お客様からの質問('93年12月号)の回答です．当件に関しましては，BJならではの高精細印字をWindows上でも利用したいというお客様よりのご要望も多いことから，現在BJ-130およびBJ-330JNM用Windowsドライバを市場に提供すべく鋭意開発中であります(リリース時期については未定)．お客様にはご迷惑をおかけし大変恐縮ではございますが，今しばらくお待ちいただきますようお願いいたします．今後とも参考ご意見をへお聞かせいただければ幸いです．

【キヤノン販売㈱広報部広報課】

もう一つの香港電脳事情

10月3日～10月6日に香港に行って来ました．本当は去年行くはずが，私が交通事故で入院(しかも自分の職場に!)したので延期になっていたのです．現地での自由時間も限られているので，家内は絹製品を，私はもちろんパソコンをと，お互いに目的をハッキリさせて行くことにしました．しかし，香港で家内に「場所はちゃんと調べてあるんでしょ」と言われ，重大な事に気付きました．そうです場所を知らなかったのです．とにかく，時間がないので，旺角まで地下鉄で行き，警官に「旺角電脳中心?」と書いたメモを見せて道を訊き，歩くこと30～40分．ありました，ありました．(なんだここだったのか!ここなら何度かビルの前を通ったのに…)さっそく中に入ってみると，安い，安すぎる!! 円高の影響もあって98派の私の心もグラグラしそうでした．ショーケースの中で目を引いたのはB5サイズ，i486SX/25MHzのノートパソコン．そのとき，私のNS/Rは色あせました．
p.s.後日，調べたらキャラベルのB5 NTシリーズでした．

【沼田市　成田】

ウイルス

11月4日㈭に職場の人から共有のPC-9821Afと，個人利用(会社での)のPC-9801DA/FAがウイルスにやられたとの報告があり，チェックソフトで確認したら，「.COM」ファイルが全部感染していた．自分のPCを確認していたら，感染はしていなかったので「ホッ」としているが，その後，チェックソフトをウイルス感染監視ソフトを組み込んで，ウイルスが入り込まないように対策をとったが，ネットワーク環境でなかったので，被害は最小限ですんだ．ウイルス感染を身近にして，ちょっとビックリしています．そこで，かなり昔にウイルスの特集を組まれた記憶があるのですが，私の個人的意見では「ウイルスのその後」と題して，最近のウイルスの傾向が知りたいので，特集を組んでもらいたいと感じています．

【横浜市　山田　誠】

コンピュータウイルスカルテ「MSU」の文字を見て，MSU-11Tの学生のレベルが上がってきたナーとナツカしく読みました(感染された方には，スミマセン！)．MSU-11T，正式名はMINDANO STATE UNIBV.-ILIGAN INSTITUTE OF TECHNOLOGYで，住所はMSU-11T，ILIGAN CITY 8801，Philippinesです．私はここで3年間，新学部設立に走り回り学部長として，スタッフに教えていまして，当時は，機材も施設も少なくて，学生も純な子供達?が多かったと記憶しています(なんといっても，16年前ですので…)しかし，そのようなINSTITUTEから世界に通用するウイルスが生み出されるとは…アア感無量です(最後に感染された方には，スミマセン…)．

【吹田市　TONY KIT】

人間ODP!?

気づいてみれば，もはや11月．センター試験は2カ月後．クラスの中にも推薦で通ったという人もいたりします．しかし自分はというと今だにドエライ成績をとっているわけで，暢気な自分も流石に慌ててきました．もし“人間ODP”なるものがあれば，実験第一号になってもいいから装着したい気分です(32ビットローカルバスで脳に直結したHDDでもいいかな?)．

【鈴鹿市　谷口哲也】

日米摩擦

あの住友化学の事故は，日米半導体協議上でも議題にのぼり，「米国メーカーに対して安定的に樹脂を供給するように努める．」ということになっているそうです．値上げの要因としては事故のせいだけではなくて，米国のメモリ需給が逼迫し，日本メーカーが国内向け分を米国に振り分けた，という経緯があり，どうやらこれが早

期の値上げにつながったのでしょう．ただ，他の樹脂メーカーが住友化学の事故で，高ビーになって，値を釣り上げているのかもしれませんケドネ．増設メモリを製造，販売しているメーカー各位様．今こそシェアを伸ばすチャンスですよ．値下げなさっては如何? Windowsの普及で咽から手が出る程，メモリを欲しがっているユーザがうじゃうじゃいますよ(かくいう私も，その内の一人です)．

【福岡市　NKYM】

“嫁の一言”

私は，11月号に掲載された“HIDE”の嫁です．事故って「体が痛くてパソコンをさわる気がしない」と書いていたくせに，HIDEは毎晩パソコンで遊んでいます．結婚1年目の嫁をほっておいて…．たまには嫁の労をねぎらってPC-486SRでも買ってくれればいいのに!! あーん，新しいパソコンがほしいよぉー

【岡山市　ゆかちゃん】

11月号で，2度目のプリンセス倶楽部への登場で，主人から正式にこの葉書の権利をもらっちゃいました．最近はモーフィングに興味があります．娘(3才)と私の顔の合成なんかやってみたいのですが，どのような機器を揃えたらよいのでしょうか．ヨロシクお願いします．

【枚方市　Hiro】

主人がパソコン通信のやり方を教えてくれたので，この間からはまっています．しかしながら，教えてもらったこと以外の操作，およびキーボードのボタンの意味は全然分かりません．少しでも勉強になればとこの本を盗み読みしましたが，チンプンカンプン．キーボードアレルギーならぬカタカナアレルギーになりそうです．グッスン．

【箕面市　由香ちゃん】

私思いますことに，さべの読者は既婚者が多いですね．よく「奥さん」とか「女房」とか「嫁さん」とか「ダンナサマ」とか目につきます．別にいいんですケドね．何かさみしいな…って思うのは私だけでしょうか．ぐすんっっ．

【広島市　まきちゃん】

TB あのー,「さべ」ではなくて「ざべ」なんですけど……．ぐすんっっ．

大変だよね

ボーナスでAT互換機を買おうと思った．しかし，2DKの家に2台のデスクトップはおけないので家を買った．おかげでAT互換機が買えなくなったうえに2,000万円もの借金をせおってしまった．

【札幌市　フミのパパ】

最近パソコンの世界がまるで面白くなくなってきた．いっときDOS/Vが話題をさらったものの，NECの反撃にあえなく失速の様相だ，というわけで今私の興味は5才の息子ともっぱらにスーパーファミコンになってしまった．

【大阪市　双子の父】

スクープ

クルマ雑誌なんかでは，スパイスクープと題して発表前の車のデータおよび写真とかを載せているが，これと同じ事をさべでできないだろうか．もちろんASCII，I/Oなどではできないから，やれるとしたらさべしかない!!

【名古屋市　ウルトラ98】

TB あのー,「さべ」ではなくて「ざべ」なんですけど……．ぐすんっっ．

レース

まささん，世界ロードレースGP2の原田優勝，本当に劇的でしたね．あの晩TV中継で観戦したあと，翌日の新聞各紙を楽しみにしていたのですが，朝日や日経は，AP共同の小さ

TWILIGHT IDLE TIME IV 黄昏劇場 とわいらいとしあた〜



Illustrated by Nobuko OKADA

な，実に小さな記事でちょろっと紹介しているだけでした．一般紙では，唯一読売が顔写真入りで，共同のニュースに自社の記事も加えてF1の結果より大きく扱っていたため，「ジャイアンツは好きじゃないが，気に入ったぞヨミウリ!!」と大きくうなずいてしまいました．世界ロードレースで，250ccクラスとはいえ，グランプリサーカスを戦い抜いてチャンピオンになるのは，並大抵の才能と努力と運ではないと思います．これがF1だったら，狂ったような報道合戦になるだろうに…．でも原田選手のためには，来シーズンもじっくりレースにとりくめる今の状況の方が，よいのかも知れませんね(レイニーは再起不能!? リハビリを始めたとかいう話しも聞かれるのですが…．レイニー抜きのGP1は考えられないなあ…)．

【神戸市　ANCHOR】

TB レイニー自身はレースへの復帰を望んでいてリハビリにはげんでいるらしいですね．

先週はF-1日本GPでした．鈴鹿市民としては当然(^_^;)TV観戦でしたが，あの雨はすごかったです．まるでセナのためのようでしたね．プロストの雨が遅すぎるのでしょうけど，プロストもワールドチャンプ決まってる事だし，もっとせめても良かったのではと思います(そんな所がプロストらしいですけどね)．それにしても，フジTVの放送は，お世辞にも上手とは言えませんね．いつもかんじんな所は，映してないし!! 今年はサーキット側の交通整理が良くて，あまり混まなかったみたいです．今年見に来た人達，ラッキーでしたね．

【鈴鹿市　魚住久紀】

新年度の標語決定

「俺たちに明日はない」

—秋葉原家電組合—

【茨城県　この手の常連】

▲▲▲

言いたい放題の上下にある▲▲▲マークが，3Dステレオグラムで見えるの私だけでしょうか．

【清瀬市　INTERSIL】

「ざべ」を1ページ目から読んでいき，目の疲れがピークに達した頃に「言いたい放題」のコーナーに突入すると，ページの上下の▲▲▲マークのせいで，妙な遠近感を感じてしまい，軽い目まいを覚えます．なんとかなりません?

【宮崎県　多田展之】

TB 今月からレイアウトを変えてみましたがいかがでしょう?

教えてくれ〜

サイリックスの新製品のCx486DX2/50，40は動作クロックが50/40MHzと書いてありますが，内部/外部，どちらのクロックなのでしょうか．内部クロックの値だとしたら，66MHz品は開発していないのでしょうか．

【府中市　エスハチ】

TB Cx486DX2ですが，50/40MHzというのは内部動作のことです．

岡田信子ファン倶楽部

11月号の岡田信子さんの作品はなんかアブないなぁ．東京に出てきたとき，何かあったのかしら?

【鶴ケ島市　五十肩】

岡田信子さんのマンガがすごく面白い．今度，特集を組んでみてはどうですか?

【福井市　渡辺慶一】

本欄では，読者の皆様のパソコンについての情報をお待ちしています．あらゆるハード/ソフトウェアに関する意見，情報，体験談など，どんなことでも結構です．また，本誌に対する感想，要望，提案なども歓迎します．巻末ハガキまたは封書でお寄せください．掲載者には図書券をお送り致します．なお，掲載にあたって，部分的に削除や表現の変更を行う場合がありますのでご了承ください．

WANTED

よりよい誌面作りのために，原稿を広く募集しています．面白いプログラム，役に立つプログラムなどできましたら，ぜひざべ誌上で発表してください．特に，下記のようなものを歓迎します．

①アプリケーションなどを使いやすくする，またはバグを回避するためのマクロ，ユーティリティ，改造方法など
②パソコン本体/周辺機器の未公開データの解析レポート
③パソコン本体/周辺機器をパワーアップするための改造方法
④諸外国のパソコン事情レポートなど
⑤ソースリストが100〜200行程度のショートプログラム

投稿される場合は，次のものを同封した上，次の宛先までお送りください．

①プログラム(ソースファイルと実行ファイル)/原稿(テキストファイル)をセーブしたフロッピーディスク．ラベルには氏名およびプログラム名/使用言語/OSを記入してください．
②次に挙げる項目を明記した用紙．投稿された方のプライバシーは固く保持しますので，必ず明記してください．

プログラム名/適応機種/適応OS(バージョンも明記)/使用言語(バージョンも明記)/必要な周辺機器/郵便番号/住所/氏名(必要であればペンネーム)/年齢/性別/職業/電話番号(自宅/勤め先)

採用され，ざべ誌上に掲載されたプログラムには，小誌規定の原稿料をお支払いします．また，ディスクサービスを行う際には印税をお支払いします．

ご投稿いただいたフロッピーディスク/原稿は，原則として返却致しません．必要な場合にはコピーをとっておいてください．また，他者の著作権侵害につながるものや，他社雑誌などとの二重投稿は固くお断り致します．

受領されたプログラムには，受け取りを確認する意味で受領通知を折り返し発送しています．なお，プログラム評価期間は3〜6カ月ほどかかります．また，評価結果によっては，機能の追加や削除などプログラムの修正，原稿の書き換えをしていただく場合があります．ご了承ください．

〈送り先〉
〒160 東京都新宿区愛住町8番地8
㈱技術評論社「ざべ編集部」宛

MONTHLY
さんば The BASIC
1994年 1月号（No.128）

FOR NEXT

発行日／1993年12月18日発行／(毎月18日発行)
通　巻　第128号
発行人　片岡　巌
編集人　冨田裕一
副編集長　松山雅明
編　集　濱中弘達／金田冨士男／加藤勝久
アシスタント　鈴木昌太／関　信浩
新見覚志／長谷川明彦
表紙デザイン　釣巻敏康
表紙イラスト　山崎のぶこ
イラスト　七戸　優／岡田信子／塩月千絵
レイアウト　ぐる～ぶざぺたん／大川デザイン事務所
写　真　広路和夫／舟喜　力／ぐる～ぶざぺたん
発行所／㈱技術評論社
〒160 東京都新宿区愛住町8番地8
編集部　☎03-3225-3295
FAX03-3225-3298
営業部　☎03-3225-2300
Jupiter-Net　☎03-3225-3291
Ganymede-Net　☎03-3225-8191
メディアセンター(広告)　☎03-3225-3481
印刷・製本／図書印刷㈱
定価／690円(本体670円)

NEXT ISSUE

'94年2月号 No.129予告('94年1月18日発売)

＜特集＞
Windowsフォントをどう選ぶか？

＜一般記事＞
対談 マイクロソフト㈱会長古川 享 vs 中村正三郎，PC-9821Bp解体新書，VIPER jet試用レポート他

※予定記事の内容，タイトルなどの一部を変更，または掲載延期することがあります．あらかじめご了承ください．

※今月号で掲載を予定しておりました『PC-9821Af解体新書』は，都合により掲載延期させていただきました．

編集後記

価格改定のお願い

いつもThe BASICをご愛読いただき，ありがとうございます．弊誌では，これまで製作費の高騰に対し，DTPの導入など経費節減に努力してまいりました．しかしながら，景気低迷の続く中，もはや内部努力だけでは対応しきれなくなりました．読者の皆様にご負担をおかけするのは心苦しいのですが，今月号より定価を690円に改定させていただくことになりました．編集部では，今後とも有用な誌面を作るため，一層の努力をしてまいります．どうか事情をおくみ取りの上，引き続きご愛読くださいますようお願い申し上げます．

TheBASIC 編集長　冨田裕一

もっと身軽に持ち運びたいという欲求が増えてきているのだろう．サブノートPCの注目度が高い．CPUも486搭載でデスクトップに肩を並べるようになってきている．ただ，どうせ携帯するのなら交信機能をつけた方が便利だろう．米国では，ポケベルの機能を持たせたHP95LXが10万台も使われているという．さて，日本ではどうなることやら．ちょっと昔にDynaBookの小ささ（大きさ？）に驚いたときがあった．来年登場するNECのサブノートは驚かせてくれるだろうか．（袋）

バブルがはじけたためマンションのオーナーが変わってしまい，マンションが取り壊しになった．まだ築3年のきれいな建物だったのに…．そんなわけで引越をしたのだが，コミックがたまりにたまって2,000冊にもなっていた．500冊は捨てて，1,000冊を古本屋へ，残りはこっそり会社に持ってきた．（まさ）

晩秋の京都を散策してきた．叡山鉄道の紅葉はもうすでに冬のものだったが，真っ赤なもみじが夕陽に照らされる様は，忘れていた郷愁を呼び起こす．てっさ・まる・カニの旨さに我を忘れ，新酒を酌み日本を祝う．
（前の後期の謎は誰もきづかなかったかな？ まっくん）

今月は，なんといいいますか，寝不足状態が続いておりまして，やる気がないわけではないんですけど，ちょっと後記に書くようなことも思い浮かばず，どうしたらいいんだろうか，なんてことを書いているうちに無事に資源の無駄遣い終了．＼(^o^)／（CTB）

減点5で，免停にリーチがかかって1年弱，びくびくしながら車で晴海や幕張へ取材に出かけていたが，ようやく減点も自然消滅．安心して車を運転できる．しかし，違反した私が悪いのは認めるが，更新で半日つぶれるというのは，どうしても納得できん!（amdo）

この号が出る頃には日本を離れ，雪があまりないというウィスラーにいる予定．バンクーバーにも3日間滞在する予定なので，北米大陸でしか買えないようなものを買ってこよう．春のオーストラリアには，きっと買ってくるようなものもないだろうし．（N.Seki）

アンケートのお願い

愛読者カードのアンケートにお答え下さい．おもしろかった記事，つまらなかった記事に関しては，以下の番号でお答えください．

郵便はがき

料金受取人払

新宿局承認

1719

差出有効期間
平成7年3月
31日まで
★切手不要★

160-00 009

(受取人)
東京都新宿区
愛住町8番地8
株式会社 技術評論社
メディアセンター
gh 資料サービス 行

フリカナ	姓	名	年齢	性別
1.氏　名				男 女
2.自　宅 住　所	〒　TEL.　(　)			
3.会　社 住　所	〒　TEL.　(　)			
4.会社名 (学校名)				
5.部署名 (学科名)	部	課 科	役職	

6.従業員数(○印を付けて下さい)	7.送付先(○印を付けて下さい)
① 10未満　② 50未満　③ 100未満 ④500未満　⑤1000未満　⑥1000以上	① 自宅　② 会社

キリトリ線

郵便はがき

料金受取人払

新宿局承認

3691

差出有効期間
平成6年10月
17日まで
★切手不要★

160-00

(受取人)
東京都新宿区
愛住町8番地8
株式会社 技術評論社
gh 編集部 行

氏　名		男・女	年令　才
住　所	〒 ☎　—　—		
職　業		ペンネーム	
使用機種			
本誌中のよく分からなかった用語(ページ数)			
Windowsの何について知りたいですか?			
プレゼント番号(P.201)	第一希望	第二希望	
Cx486SLC/DLCを持って	いる　いない	主な使用メディア	5"FD　3.5"FD

キリトリ線

ざべ '94年1月号　愛読者カード

❏今月号で興味を引かれた記事は何でしたか（P.208の記事番号でお答えください）

❏また，興味がなかった記事は何でしたか（理由も）

❏今後取り上げてほしい記事，特集は何ですか（具体的に）

❏読者投稿欄　あなたのパソコンに関する情報，体験談など，また，質問や本誌に対するご意見などをお寄せください．

キ リ ト リ 線

資料請求カード ざべ　1月号

（注）このカードの有効期限は1994年1月15日です。

資料請求カード　ざべ　1月号
恐れ入りますが，下記の該当する番号をご記入下さい．

職種

1. コンピュータハード技術者　2. コンピュータソフト技術者（含プログラマ）
3. その他の技術者　4. 企業経営者　5. 管理職　6. 商工サービス・自営
7. 研究・開発　8. 事務職　9. 販売職　10. 専門職（弁護士・医師など）
11. サービス職　12. 教職　13. 学生　14. その他（　　　　）

産業分野

1. コンピュータ関連　2. 農林業・漁業・鉱業　3. 建設業　4. 不動産業
5. 商社・卸売・小売業　6. 金融・証券・保険業　7. 運輸業　8. サービス業
9. 学校・教育関係　10. 官公庁・研究機関　11. 製造業　12. 医療関係
13. 放送・出版関係　14. その他（　　　　）

パソコン用途　（2つまでご記入下さい）

1. ハード開発　2. ソフト開発　3. 科学技術計算　4. 数値処理統計　5. 事務処理
6. 趣味　7. その他（　　　　）

主に使用するソフト　（2つまでご記入下さい）

1. ワープロ　2. 表計算　3. データベース　4. 業務ソフト統合型　5. CAD
6. グラフィックス　7. 開発研究ツール　8. プログラミング言語　9. 音楽
10. ゲーム　11. その他（　　　　）

今後購入を考えている周辺機器

1. ハードディスク　2. モデム　3. プリンタ　4. その他（　　　　）

主に使用するコンピュータ

メーカー名	機　種　名

資料請求欄（請求No.の欄に資料請求番号を記入し1～3のご希望の項目に○印を付けて下さい.）

希望項目　請求No.										
1. 購入希望	1	1	1	1	1	1	1	1	1	1
2. 直接説明を聞きたい	2	2	2	2	2	2	2	2	2	2
3. カタログ送付希望	3	3	3	3	3	3	3	3	3	3

キ リ ト リ 線　　キ リ ト リ 線

チャンスです。

下取交換表

※数字は差額です。

下取機種	新品機種 9801BA2-U2	9821Ap2-U2	PC-9821NE	PC-486HX3	9801BX2-U2	9801BS2-U2	9801NXC	PC-486SE2	PC-486SR2
PC-9801BX-U2	183,000	266,000	298,000	208,000	62,000	108,000	205,000	45,000	98,000
PC-9801DA-U2	186,000	269,000	301,000	211,000	65,000	111,000	208,000	48,000	101,000
PC-9801FA-U2	163,000	246,000	278,000	188,000	42,000	88,000	185,000	25,000	78,000
PC-9801NA-C	53,000	136,000	168,000	78,000	———	———	75,000	———	———
PC-9801NC	103,000	186,000	218,000	128,000	———	28,000	125,000	———	18,000
PC-9801NSE	175,000	258,000	290,000	200,000	54,000	100,000	197,000	37,000	90,000
PC-9801NSE40	166,000	249,000	281,000	191,000	45,000	91,000	188,000	28,000	81,000
PC-9801NSR	143,000	226,000	258,000	168,000	22,000	68,000	165,000	5,000	58,000
PC-9801NST	163,000	246,000	278,000	188,000	42,000	88,000	185,000	25,000	78,000
PC-9801RA21	191,000	274,000	306,000	216,000	70,000	116,000	213,000	53,000	106,000
PC-9821Ap-U2	13,000	96,000	128,000	38,000	———	———	35,000	———	———
PC-9821As-U2	63,000	146,000	178,000	88,000	———	0	85,000	———	———
PC-386NAR2	163,000	246,000	278,000	188,000	42,000	88,000	185,000	25,000	78,000
PC-486GR2	153,000	236,000	268,000	178,000	32,000	78,000	175,000	15,000	68,000
PC-486GR3	143,000	226,000	258,000	168,000	22,000	68,000	165,000	5,000	58,000
PC-486NAS2	138,000	221,000	253,000	163,000	17,000	63,000	160,000	0	53,000
下取無	253,000	336,000	368,000	278,000	132,000	178,000	275,000	115,000	164,000

内税　内税

下取機種	新品機種 PR1000E4	LBP-A404G2	LP-1500	BJ-220JC
PC-PR101-TN103	79,000	91,000	75,000	38,000
PC-PR201-45	72,000	84,000	68,000	31,000
PC-PR201-60	69,000	81,000	65,000	28,000
PC-PR201G	80,000	92,000	76,000	39,000
PC-PR201GS	79,000	91,000	75,000	38,000
AP-1000PC	82,000	94,000	78,000	41,000
AP-700	81,000	93,000	77,000	40,000
AP-900PC	83,000	95,000	79,000	42,000
VP-1350PC	85,000	97,000	81,000	44,000
VP-2600PC	74,000	86,000	70,000	33,000
BJ-10Vカスタム	82,000	94,000	78,000	41,000
BJ-10Vセレクト	80,000	92,000	76,000	39,000
BJ-15V	73,000	85,000	69,000	32,000
DeskJet 505J	72,000	84,000	68,000	31,000
下取無	102,000	114,000	98,000	61,000

内税

中古価格表

PC-9801BX-U2	89,000	PC-9801UX21	55,000
PC-9801DA-U2	87,000	PC-9801VX21	35,000
PC-9801FA-U2	115,000	PC-9821Ap-U2	288,000
PC-9801N	49,000	PC-9821As-U2	235,000
PC-9801NA	159,000	PC-PR201G	38,000
PC-9801NAC	238,000	PC-386NAR2	109,000
PC-9801NC	185,000	PC-386NWR2	87,000
PC-9801NS20	79,000	PC-486GR2	125,000
PC-9801NSE	96,000	PC-486GR3	135,000
PC-9801NSE40	110,000	PC-486NAS2	139,000
PC-9801NSR	129,000	AP-1000PC	26,000
PC-9801NST	109,000	BJ-10Vカスタム	28,000
PC-9801NV	64,000	BJ-10Vセレクト	29,000
PC-9801RA21	85,000	DeskJet505J	42,000
PC-9801RX2	49,000	外付け100Mハードディスク	28,000
PC-9801RX21	51,000	外付け40Mハードディスク	22,000

内税

《価格表注意》ケ付＝ケーブル付
※＝単品での買取り，下取り，販売はいたしません。

宣伝商品在庫率90%以上（中古品は除く）　今すぐお電話を!

お申し込み

- **通販**(下取交換含む・無休) **03-3739-2911**
- **来店**(下取交換含む) **03-3739-8628**
- **中古販売専用 03-3739-3069**
- **在庫、税込価格、送料をご確認ください。**
  (送料・1ケロ・本州￥1,000、北海道・四国・九州￥1,500、沖縄・離島は応談)

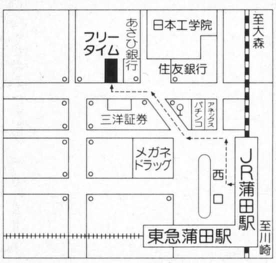
通販の方の振込先

●銀行に備えつけの振込用紙に下記のようにご記入ください。

振込銀行	種目	口座番号
富士銀行蒲田(カマタ)支店	当座	24143

口座名義	電話	住所
フリータイム(株)	03-3739-2911	東京都大田区西蒲田5-27-13

※必ず電信扱いでお振込みください。ご入金確認後に発送いたします。振込手数料はお客様負担です。



- 在庫品は即日発送、即日お持ち帰り可能です。
- 下取交換の方は、下取品(付属品等含む)宅配便で、着荷、入金(差額、送料)が確認されしだい発送となります。
- ショッピングクレジット希望の方は申し込み用紙をお送りします。
  クレジット成立次第発送いたします。
  中古品等一部クレジット販売をお断りする商品があります。
- クレジットカードは扱っていません。

クレジット金利率				
12回払い	24回払い	36回払い	48回払い	60回払い
5.4%	11.3%	15.8%	20.9%	26.9%

フリータイム(株)
AM10:00～PM7:00火曜日定休(通販無休)
〒144 東京都大田区西蒲田5-27-13

LIFEBOAT

他人のソースを読みたくない時の必須ツール

大好評

C View™

（シー・ビュー）

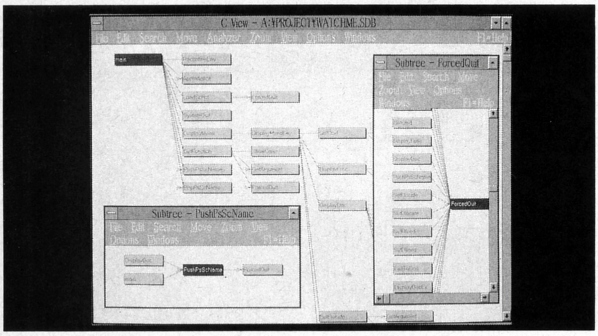
C言語 ソースコード ナビゲータ

標準価格 ¥48,000

CViewはC言語専用のナビゲーション環境を提供するリバース・エンジニアリング・ツールです。多数のソースファイルからなるアプリケーションプログラムを自動解析し、構造チャートとして視覚化します。チャート上の任意の箇所をマウス・クリックするだけで、関連する関数を抽出したりエディタに移ったりのナビゲーションが自由自在。プログラムの構造把握には、ソースコードを書いた本人でさえ時間がかかるもの。プログラミングのスムーズな継続をCViewが可能にします。

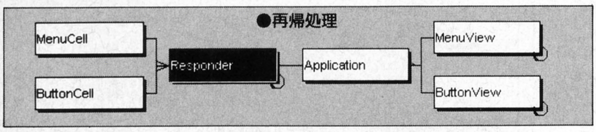
構造チャートを自動生成

■CViewはC言語で書かれたソースプログラムから構造チャートを生成します。ボックスで表された関数と、その連結を示すラインによってプログラムの連鎖構造を視覚的に捉らえることができます。



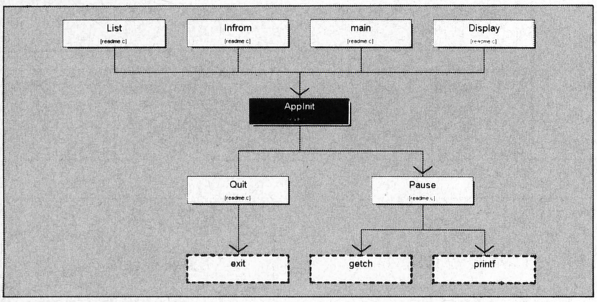
再帰処理も解析・表示

■関数内で自分自身を呼び出す再帰処理や、別の関数をバイパスした後に自分に戻ってくる間接再帰などの複雑なプログラム構造も的確に判断できます。



呼び出し関数の抽出

■CViewはある関数が呼び出した他の関数や、他の関数から呼ばれたある関数の連鎖関係を切り出します。これによって、特定の関数を変更した時に影響を及ぼす範囲を限定することができます。



View & Editの快適編集

■インターナルエディタによって、関数間の流れを確認しながらインタラクティブにソースファイルの編集を行なうことができます。関数を表すボックスを選択するだけで、ソースファイル中の関数が定義された位置を自動的に表示します。

関数間インターフェースを簡単参照

■構造チャートに表示された各ボックス(関数)を選択することによって、定義された関数プロトタイプやその関数に与えられたコメントを表示することができます。これによって、エディタに入る前に大まかな入力と出力の処理内容を得ることができます。

プリンタ出力で仕様書に

■CViewで解析した構造チャートは、そのままプリンタへ出力することができます。プログラム仕様書としてそのまま利用できます。

C Viewの特徴

●プログラムを変更した際の影響範囲を調べる。

ある関数を修正すると、その関数を呼び出していたすべての関数が連鎖的に影響を受けます。整合性をチェックし、悪影響を防がなければなりません。しかし、影響を受ける関数を特定することは開発者本人でさえも困難です。C Viewはこの連鎖関係を自動解析し、影響範囲を特定します。

●開発者不在のソースコードを全体的に分析する。

開発者以外のメインテナンス要員が行なう修正は、殆どの場合、場当り的な修正に終始します。これは、第三者がプログラム全体の構造を把握することが事実上困難であることに起因します。しかし、第三者であっても、C Viewによってプログラム全体の流れを簡単に得ることが可能です。

●大規模プログラムから修正範囲を切り出す。

C Viewで解析すれば、main関数につながっていない関数がひと目で分ります。そのような実際には使われていない汎用関数や、気にする必要のない標準関数を表示しないようにも指定できます。

●構造を把握しながらエレガントなプログラムを開発する。

C Viewは、従来のテキストエディタのみによるプログラミングと異なり、視覚的にプログラム構造を把握しながら、インターナルエディタによる開発が行なえます。構造チャートと対比しながらプログラミングすることは、ロジック的な誤りを未然に防ぎ、エレガントなプログラム開発を促します。

【対応OS】MS-Windows 3.0/3.1（スタンダードまたはエンハンスドモード）【対応機種】NEC PC-9800シリーズおよび各社DOS/V仕様機 【必要メモリ】4MB以上
【対応Cコンパイラ】Microsoft C/Turbo C/ANSI/K&Rなどの主要文法に対応
＊ご注文の際はメディアタイプ(3.5インチ あるいは 5 インチのいずれか)および対応機種（PC-9800版あるいはDOS/V版)をご指定下さい。

MICROSOFT® WINDOWS™ COMPATIBLE

●C Viewはお近くのパソコンソフト取扱店でお求め下さい。

株式会社 ライフボート
〒101 東京都千代田区神田錦町3-6
PHONE:03-3293-4711(代) FAX:03-3293-4710

あなたのパソコン・ライフを バック・アップします

あなたはパソコンを買ってから
「しまった」と
思いませんでしたか？
来年は「もっとすごい」機種が
出るかも知れません
考えて見ませんか？

「お金と資源の無駄使い」

CPU交換

《¥34,800》

CPU（486SLC）込み

あなたのパソコンに使われているCPUを交換して処理スピードを上げます。どのようなメーカー、機種でもCPUを交換出来ますが残念ながら特殊な機種、型の古い機種は料金が少々高くなります、上記金額は386sxを486SLC（25MHz）に交換した場合です。

パソコン改造

クロックアップ ノート型タイプ ¥10,000	RGB出力 ¥17,000	キャッシュ・コントロール 3.5/5 FDD ¥ 5,000

コプロ（FPU）
FPU込み
¥19,000

お急ぎのお客様のため、即日による改造作業も実施致しております。またオリジナルな改造やユニークなアイデア改造がございましたら、当社が実現致します。

お年玉キャンペーン

［先着94名様まで］

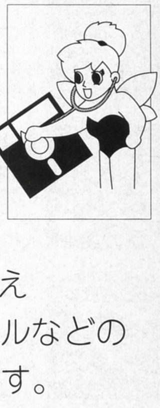
CPU交換 ¥30,500

当社まで直接ご連絡ください、受付Noをお知らせします、受付Noがないと通常料金になるのでご注意下さい。

基本的に製造メーカー、機種、作業内容を問いません。パソコンの改造からシステムの構築まで幅広くお受け致します。
明細は下記問い合わせ先までご連絡下さい、パンフレット等送付させて頂きます。

日常支援

クリーニング 1件（1ユニット） ¥ 5,000	ウィルス・チェック 基本料金 ¥ 5,000
インストール 1件（1ユニット） ¥ 3,000	データ・救急 基本料金 ¥15,000



総合的なパソコン・ライフを考え
ウィルスチェックやインストールなどの
日常的な作業をお手伝い致します。

問い合わせ先

岡崎電機 株式会社東京事務所
〒140 東京都品川区南大井3-5-10
カーサグランデ2F（207号）
TEL(FAX) 03-3767-0998
受付 9:00～17:00（日祭日を除く）

パソコン通信

PC-VAN HPH65334
NIFTY KHB00257

資料請求No. 11


A.I.SOFT.INC.
Leading Software by A.I. Technology
圧倒的なコストパフォーマンスと、
自由度の高い開発環境。
これが今度のUNIX。
Welcome to UNIX
98&AT
汎用ハード
CPUが386以上のお手持ちのマシンでUNIXが使用できます。PC-98シリーズ、PC/AT互換機のお好きな方を選択いただけます。
SVR4
本物
業界標準のSystem/V Release 4.0の本格カーネルを採用しています。
cc&gcc
開発環境
標準で2つのコンパイラ(標準ccとgcc 2.4.5)とライブラリ群を装備し、ソフトの開発・移植、XやLAN対応のプログラムが作成可能です。
X11R5
最新GUI
X Window SystemにはX11R5を採用し、最新のS3アクセラレータボード(最大1024×768:256色)に対応することで、ワークステーション並みの速度を実現。
MANUAL
英語/日本語
英語のフルオンラインマニュアルの他に、主要コマンドは日本語マニュアルを付けました。
パーソナルUNIXシステム for 98 & for AT
PANIX
パニックス
Ver.3.0
本体価格48,000円(消費税は含まれておりません)

雑誌04095-01

T1004095010691

Printed in Japan

定価690円(本体670円)